# Modelling and revealing intelligence in complex biological systems: a regard from genetic circuits to neuronal networks

*Zonglun Li*

A dissertation submitted in partial fulfillment

of the requirements for the degree of

**Doctor of Philosophy**

of

**University College London**.

Department of Mathematics

University College London

October 10, 2024

I, Zonglun Li, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.

# Abstract

Numerous real-world phenomena and challenges require a new way of thinking from a systems perspective and this new approach is broadly called complex systems. Instead of having to break down a system into its individual components and studying their respective dynamics and contribution to the entire system, complex systems adopt a more collective approach with the emphasis on the whole rather than the sum of it, while sometimes the individual components may still be of interest. Among them, biological systems have attracted growing attention in recent years and are reckoned able to demonstrate a wealth of intelligence of different types which has remained untapped by humans. Therefore, the thesis is dedicated to the advancement that has been made during my PhD in revealing the intelligence that the systems can exhibit, ranging from molecular circuits to neuronal networks with the aid of mathematical and computational models. We will first leverage Hill equations to investigate the two advanced properties for associative learning that we newly proposed in the context of genetic circuits. Later, we will focus on the various forms of intelligence that neuronal networks can enable. It comprises the investigation of short-term memory in the presence of astrocytes, information processing and the disorder of the network, and classifying time series inputs.

# Impact Statement

The thesis comprises the breakthrough that has been made to better reveal the intelligence that complex biological systems can potentially possess and demonstrate using mathematical and computational models.

In Chapter 2, we manage to characterize two advanced properties of associative learning that can be potentially exhibited by synthetic genetic circuits using mathematical models. To this end, we propose two advanced properties, reinforcement effect and forced dissociation, that can be considered in the design of new genetic circuits so as to unleash the untapped potential of synthetic biology. Therefore, we demonstrate that the Fernando's model can display the reinforcement effect but fails to display the forced dissociation, while we also construct a new model that can meet both requirements at the cost of several limitations.

In Chapter 3, we arrive at a plausible hypothesis of the impact of astrocytic coverage of synapses on the short-term memory using a neuron-astrocyte network. Working memory refers to the capability of the nervous system to selectively retain short-term memories in an active state and besides, growing evidence has suggested that astrocytes exhibit diverse coverage of synapses which are considered to participate in neuronal activities. Hence, we leverage a computational neuron–astrocyte model to study the short-term memory performance subject to various astrocytic coverage and we demonstrate its susceptibility along with varying spatial frequencies.

Biological systems also possess considerable potential in learning new patterns and dynamics. Among all machine learning challenges, time series classification (TSC) has gained increasing attention due to its wide application and impact. In

Chapter 4, we develop two reservoir computing methods, Diff-ESNs and Interp-ESNs, that can efficiently deal with a wide spectrum of challenging TSC tasks. Among 33 UCR time-series benchmark datasets, the Diff-ESN outperforms the gold standard dynamic time warping (DTW) method on 23 datasets whilst achieving comparable results on the remaining 10 datasets. As for more irregular time series data, the Interp-ESN can be used to achieve desirable performance and enable flexible forecasting.

In Chapter 5, we harness the core structure of liquid state machines, another type of reservoir computing, to investigate the propensity of entropy change in response to the external stimulation. To this end, a new biophysical formalism has been proposed to properly simulate the dynamics of a pseudo-brain in the presence of external input with biological plausibility. Most importantly, we provide a reasonable explanation for the reduced brain entropy in some studies from the modelling perspective. Besides, we reveal the propensity of entropy change within the networks of different sizes and shed more light on these observations by delving into the dynamics of the networks.

# Acknowledgements

I acknowledge Prof. Alexey Zaikin and Dr. Oleg Blyuss for their academic and mental support throughout my PhD study. My thanks also go to Susanna Gordleeva, Yuliya Tsybina, Luis Abrego and Nikita Sushentsev for the helpful discussions during the collaboration. I am also grateful to the participants of the Mathematical Biology Meetings for this ongoing opportunity to broaden my horizon. Besides, I will miss all of my PhD colleagues and not least the laughter and joy in KLB that kept my chin up all year round. Equally importantly, a big shout out to the London Go (Weiqi) community, especially the London Go Centre, that makes me feel at home outside academia. Last but not least, I owe a million thanks to my close relatives and girlfriend Wanqing, who are always with me in my highs and lows.

# UCL Research Paper Declaration Form (1)

1. **1. For a research manuscript that has already been published** (if not yet published, please skip to section 2)**:**

   (a) **What is the title of the manuscript?** An account of models of molecular circuits for associative learning with reinforcement effect and forced dissociation

   (b) **Please include a link to or doi for the work:** https://www.mdpi.com/1424-8220/22/15/5907

   (c) **Where was the work published?** Sensors

   (d) **Who published the work?** MDPI

   (e) **When was the work published?** 7 August 2022

   (f) **List the manuscript's authors in the order they appear on the publication:** Zonglun Li, Alya Fattah, Peter Timashev, Alexey Zaikin

   (g) **Was the work peer reviewed?** Yes

   (h) **Have you retained the copyright?** Yes

2. **For a research manuscript prepared for publication but that has not yet been published** (if already published, please skip to section 3)**:**

   (a) **What is the current title of the manuscript?**

   (b) **Has the manuscript been uploaded to a preprint server 'e.g. medRxiv'?**

**If 'Yes', please give a link or doi:**

(c) **Where is the work intended to be published?**

(d) **List the manuscript's authors in the intended authorship order:**

(e) **Stage of publication:**

3. **For multi-authored work, please give a statement of contribution covering all authors** (if single-author, please skip to section 4)**:** Supervision: A.Z.; conceptualization: A.Z. and Z.L.; model derivation: A.F. and Z.L.; numerical simulation: Z.L.; writing: Z.L., A.Z., A.F. and P.T.

4. **In which chapter(s) of your thesis can this material be found?** Chapter 2

# UCL Research Paper Declaration Form (2)

1. **1. For a research manuscript that has already been published** (if not yet published, please skip to section 2)**:**

   (a) **What is the title of the manuscript?** Impact of Astrocytic Coverage of Synapses on the Short-Term Memory of a Computational Neuron-Astrocyte Network

   (b) **Please include a link to or doi for the work:** https://www.mdpi.com/2227-7390/10/18/3275

   (c) **Where was the work published?** Mathematics

   (d) **Who published the work?** MDPI

   (e) **When was the work published?** 9 September 2022

   (f) **List the manuscript's authors in the order they appear on the publication:** Zonglun Li, Yuliya Tsybina, Susanna Gordleeva, Alexey Zaikin

   (g) **Was the work peer reviewed?** Yes

   (h) **Have you retained the copyright?** Yes

2. **For a research manuscript prepared for publication but that has not yet been published** (if already published, please skip to section 3)**:**

   (a) **What is the current title of the manuscript?**

   (b) **Has the manuscript been uploaded to a preprint server 'e.g. medRxiv'?**

**If 'Yes', please give a link or doi:**

(c) **Where is the work intended to be published?**

(d) **List the manuscript's authors in the intended authorship order:**

(e) **Stage of publication:**

3. **For multi-authored work, please give a statement of contribution covering all authors** (if single-author, please skip to section 4)**:** Supervision: A.Z.; conceptualization: A.Z. and Z.L.; methods: S.G. and Z.L.; numerical simulation: Y.T. and Z.L.; writing: Z.L., A.Z., S.G. and Y.T.

4. **In which chapter(s) of your thesis can this material be found?** Chapter 3

# UCL Research Paper Declaration Form (3)

1. **1. For a research manuscript that has already been published** (if not yet published, please skip to section 2)**:**
   (a) **What is the title of the manuscript?**
   (b) **Please include a link to or doi for the work:**
   (c) **Where was the work published?**
   (d) **Who published the work?**
   (e) **When was the work published?**
   (f) **List the manuscript's authors in the order they appear on the publication:**
   (g) **Was the work peer reviewed?**
   (h) **Have you retained the copyright?**
2. **For a research manuscript prepared for publication but that has not yet been published** (if already published, please skip to section 3)**:**
   (a) **What is the current title of the manuscript?** Novel efficient reservoir computing methodologies for regular and irregular time series classification
   (b) **Has the manuscript been uploaded to a preprint server 'e.g. medRxiv'?**
   **If 'Yes', please give a link or doi:** No

(c) **Where is the work intended to be published?** Nonlinear Dynamics

(d) **List the manuscript's authors in the intended authorship order:** Zonglun Li, Andrey Andreev, Alexander Hramov, Oleg Blyuss, Alexey Zaikin

(e) **Stage of publication:** Not submitted

3. **For multi-authored work, please give a statement of contribution covering all authors** (if single-author, please skip to section 4)**:** Supervision: A.Z., O.B. and A.H.; conceptualization: Z.L., A.Z. and O.B.; methods: Z.L.; numerical simulation: Z.L. and A.A.; writing: Z.L., A.A.

4. **In which chapter(s) of your thesis can this material be found?** Chapter 4

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Review and overview

In this day and age, numerous real-world phenomena and challenges require a new way of thinking from a systems perspective, ranging from ecosystems to urban traffic, and we name this new approach complex systems [1–3]. The complex system approach generically features the partial or global collective behaviors that emerge from an intertwined network where the individual status of the agents is no longer of great importance. In particular, mathematical modelling of complex systems attempts to unravel the underlying properties that underpin the global dynamics, reproduce the observed phenomena and suggest novel behaviors that are yet unknown to the public [4–6]. Among them, biological systems have attracted growing attention in the last few decades due to their broad potential implication for managing the ecological environment and medical treatment, etc [7–10]. Spanning from single cells to the whole ecosystem, the biological components coordinate in an either cooperative or competitive way in order to sustain the normal function of the living organisms. Therefore, interdisciplinary approaches have been widely used to expand the scope of the field and enhance our understanding of the systems at different hierarchical levels.

Intelligence refers to the ability of a unit or a system to exhibit behaviors such as decision making, comprehension, generalization and memory formation that attain or transcend beyond the intellectual level of humanity [11]. Nowadays, it is widely accepted that biological systems possess a wealth of intelligence of different types and their potential is largely untapped by human beings [12–18]. Even

though intelligence is a fairly abstract concept and it is nearly impossible to give a precise definition to each form of intelligence, it is still worthwhile to dig into this complicated field from a numeric perspective for several reasons. Firstly, it can help us better understand the formation of self-awareness of humans and the extent to which the other living systems have developed; secondly, it can help bridge the gap between the biological intelligence and the artificial intelligence and a case in point is the latest development in neuromorphic computing [19–21]; thirdly, it may provide theoretical foundations for the design of engineered systems that can potentially revolutionize the treatment of various diseases. As a result, in this thesis, we will present the advancements of revealing and modelling the emergent intelligence from the biological systems that have been made during my PhD. Specifically, the biological systems that we are mainly concerned with are genetic and neuronal networks and furthermore, the primary focus revolves around investigating the emergence of intelligence by virtue of the temporal and spatial variation of the components in these systems. Indeed, spatial and temporal variation are universal phenomena and reckoned to underpin the subsistence of many species. For instance, [22] quantified the pattern of the spatial variation within a bird species and discovered that the species was orders of magnitude more abundant in a few locations than others; [23] examined the spatial variation of the refractive index in liver cells and revealed that only the nucleus and the membrane of the cell yield a visibly different phase shift relative to the mean value of the cell; [24] assessed the spatial as well as the temporal variation in plant hydraulic traits and their impact on vegetation subject to climate change. In [25], visible temporal and spatial differentiation was observed among the endangered Lake Saimaa seals in a time span of several decades. All these findings may point to the necessity of some structural formulation of intelligence emergence facilitated by the information transmission between the components in a biological system owing to their spatial and temporal variation in concentration. In the next few paragraphs, we will define the precise research questions that we attempt to answer and introduce the fundamental concepts and terminologies that underpin our methodologies in the meantime.

Synthetic biology is an interdisciplinary field that combines expertise from biology, chemistry, and engineering to design and construct new devices or systems [26–29]. They are normally expected to perform specific functions or produce desired products in an attempt to address the challenges that arise from medicine, agriculture and environment, etc. There are many key components in synthetic biology and here the focus is on the cellular engineering as this is the part that is the most relevant to the mathematical modelling that will be covered later. The cellular engineering involves the construction of synthetic genetic circuits and regulatory elements so as to control gene expressions and cellular behaviors. Genetic circuits have numerous applications and many studies have demonstrated the potential of them to carry out some specific tasks which are reckoned to require a certain level of intelligence by human standard [30–35]. They are composed of interconnected genetic and regulatory elements, such as genes, promoters and proteins, that interact with each other to accomplish some specific functions. Promoters are DNA sequences that regulate the onset of transcription as well as the extent to which a gene is expressed. Regulatory elements are anything that can modulate gene expressions through the interactions with promoters. The most commonly used elements are proteins such as transcription factors, activators and repressors. A simple example is given in Figure 1.1. The transcription factor binds to the promoter that initiates the transcription of the gene, and a product is produced as a result. In mathematics,

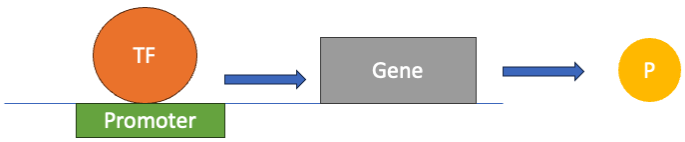



**Figure 1.1:** A simple genetic circuit. TF: transcription factor, P: product.

the aforementioned reactions are conventionally described by the Hill equation:

$$O_b = \frac{T^n}{K^n + T^n} \tag{1.1}$$

Here $O_b$ is the fraction of the operator concentration that is bound by the tran-

scription factor, $T$ is the concentration of the specific transcription factor, $K$ is the Hill constant and $n$ is the Hill exponent. More details about the origin of the Hill equation along with the derivation of the translation equation as well as biological background can be found in Appendix A.

Associative learning is one of the most fundamental principles that underlie our psychology and physiology [36–39]. It is aimed to investigate and depict how the association of certain stimuli takes place with specific responses as well as their implication for memory and decision-making. One of the main types of associative learning is the classical conditioning, featured by the renowned Pavlov's dog experiment. The experiment can be summarized as through repeated pairings of the food presence with the bell's ring, the dog would eventually be able to make the connection between the food and the ringing. Next time, when only the bell ringed, the dog would still salivate even without food presence. An active area in synthetic biology is to design and implement novel genetic circuits with the property of associative learning [35, 40, 41]. However, the current development is restricted to the formation of the association and the discussions around the properties beyond this is absent. Hence, in Chapter 2, these properties will be accounted for comprehensively and mathematical models will be constructed to furnish a theoretical basis as well as generate simulation results. In parallel, we will demonstrate that the temporal variation in the concentration of proteins involved in the circuit are behind the realization of these properties.

Neurons are the basic processing units in the brain and the most fundamental building blocks of the communication system. Neurons in the brain form intricate circuits or networks and the transmission of electrical or biochemical signals from one neuron to another is permitted by the physiological junction, synapse [42–44]. Enormous evidence has revealed that neurons play an imperative role in the formation of intelligence of various types that is instrumental to the subsistence of humans and other animals. It includes but is not limited to plasticity, learning and decision making, sensory and information processing and memory formation [45–52].

Astrocytes are star-shaped glial cells that have been found prevalent in the

brain. While neurons are often considered the primary cells in the brain, in recent years, astrocytes have been increasingly recognized for their indispensable contribution to the normal functioning of the brain [53–56]. It is now widely believed that astrocytes are not only the structural guardians of the neurons and the nervous system, but also actively involved in brain activities such as the formation of the blood-brain barrier, synapse formation and elimination, participation in neuronal signaling, etc. In this work, we will focus on the contribution of astrocytes to the neurotransmission between pre- and postsynaptic neurons. The process is enabled by the tripartite synapse [57–59]. It is a specialized type of synapse in the sense that the transmission is facilitated by the presence of the perisynaptic astrocytes. A schematic diagram is shown in Figure 1.2. The neurotransmitters (blue dots) released from the terminal of the presynaptic neurons, often as a result of the occurrence of action potentials, bind to the receptors on the astrocyte and initiate the release of gliotransmitters (yellow dots). The gliotransmitters will thereafter bind to the receptors on the terminal of the postsynaptic neurons and the signals have been bridged from one side to another. More detailed mechanisms along with the mathematical models will be described in Chapter 3.

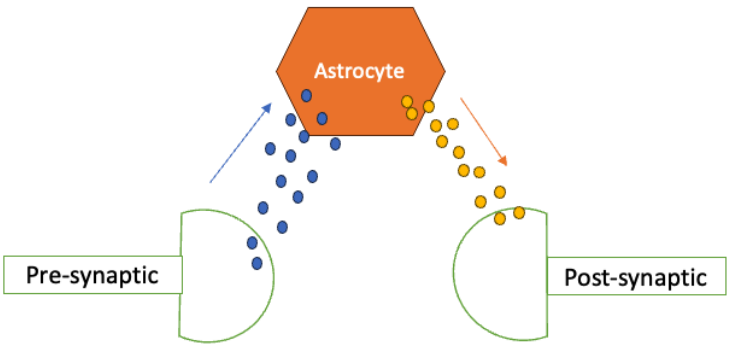



**Figure 1.2:** A schematic diagram of the tripartite synapse

Working memory is reckoned prevalent in the nervous system and is responsible for the temporary storage and retrieval of information crucial for various cognitive tasks [60–62]. It creates a temporary workspace where transient information can be held and processed to inform actions such as decision-making or learning. The duration of working memory is typically in seconds. It is often regarded as

an extension of the term short-term memory. Even though a precise boundary between the working memory and the short-term memory has never been drawn, one can generally interpret working memory as a more dynamic and complex system while short-term memory is more often applied to a more static scenario. For instance, recalling the picture shown to you 5 seconds ago may be categorized as short-term memory whereas recalling the exact sequence of motions may be considered in relation to working memory. Since we will only implement a simply vision task, short-term memory will be used for the most part to avoid providing misleading information. In [63–66], in silico neurn-astrocyte models have been developed to investigate the formation and the performance of the working memory with the assistance of astrocytes. Nevertheless, the coverage of the astrocytes remains unchanged and the alteration in spatial frequencies of the input patterns has been left out of discussion. Therefore, in Chapter 3, we will study the impact of astrocytic coverage of synapses on short-term memory in the presence of varying spatial frequencies of the input signals using a neuron-astrocyte computational model. Essentially, the spiking frequencies of individual neurons and the concentrations of ions in individual astrocytes can be viewed as the main source of spatial variation in the network. Moreover, the alteration of astrocytic coverage of synapses can be regarded as the variation in the spatial pattern given that the density of astrocytes will be adapted accordingly.

Reservoir computing is a specific type of recurrent neural networks and has been broadly applied in machine learning and neuroscience to deal with time series data [67–70]. It normally consists of three layers: the input layer, the reservoir layer and the output layer. The output layer is essentially indispensable when it comes to machine learning tasks but can be removed if one only wants to study the dynamics in the reservoir layer. As a matter of fact, if one removes the output layer, it is effectively a simplified representation of how the brain receives input signals and processes the acquired information within it, as shown in Figure 1.3.

As for the machine learning tasks such as time series prediction and classification, the precision is of the top priority and the biological accuracy as a conse-

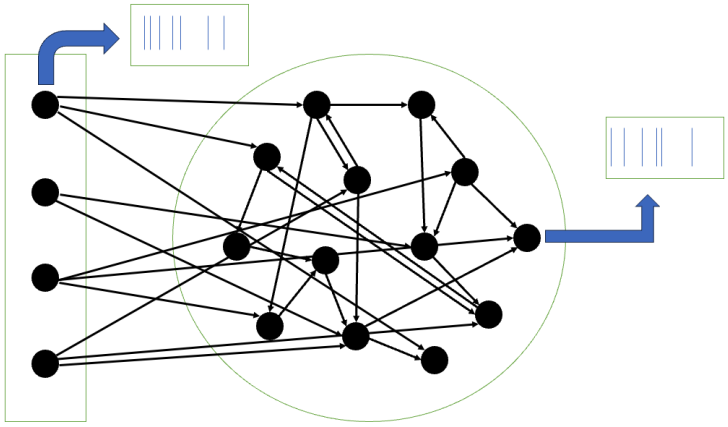


**Figure 1.3:** A schematic diagram of the first two layers of reservoir computing

quence, may need to be compromised. Echo state networks (ESNs) are a type of RC where the sigmoid functions are used as the activation functions for the neurons in the reservoir layer to permit information transmission and computational efficacy within the network [71]. The role played by the sigmoid activation function is to ensure the echo state property (ESP) which will be explained in due course. The idea is to map the input time series to a high-dimensional recurrent neural network such that the intrinsic dynamics can be better characterized. It is attributed to the temporal and spatial variation in the activation values in the reservoir layer as a result of the weights in the input and the reservoir layer, the values of which are randomly assigned. The training is conducted by adjusting weights in the output layer using statistical machine learning techniques in the hope that the right connections can be established between the input and the output. Up until now, ESNs have demonstrated colossal success in sequential predictions but the potential of dealing with time series classification (TSC) tasks is still to be discovered. In Chapter 4, we will develop efficient approaches that can perform TSC for an array of time series datasets.

Information processing in the brain involves a series of intricate steps that allow us to perceive, store and analyze information. The sensory receptors discover the external stimuli and integrate information into the nervous system. In particular, information theory and the brain are considered interconnected in mysterious ways, not least because the brain can be regarded as a complex information processing

system as well as that the spiking patterns resemble some coding theories [72–76]. One of the most central concepts in information theory is the entropy. Entropy is a quantitative measure of the amount of disorder, uncertainty, or surprise in a system and in the backdrop of brain science, some researchers believe that it in a way, reflects the diversity in the firing patterns and the efficiency in information transmission [77–84]. Nowadays, brain stimulation techniques are extensively employed to modulate the brain activity with the aim of treating neurological diseases and unveiling the mystery of brain functions [85–88]. However, apart from a few isolated studies [89–91], a general understanding of how the entropy in the brain is affected by the external stimuli is still deficient, particularly from the modelling viewpoint. Therefore, a preliminary analysis of this crucial relationship using mathematical approaches will constitute the main content of Chapter 5.

Liquid state machines (LSMs) are another type of RC where the units are biological neurons, as in more emphasis is put on biological plausibility [92–94]. The liquid reservoir is composed of numerous recurrent spiking neurons and they are described by biophysical neuronal models in silico. It is also preferable that the input of the time series data to the reservoir layer can be converted to spike trains using some specific neural encoding methods. Consequently, the reservoir can be seen as the brain in miniature and the dynamics of the brain susceptible to the external stimulation can be studied numerically. With that being said, we should also point out that the LSMs can be equally adopted to perform machine learning tasks through the training of the output layer, while facing more difficulty in terms of efficacy as compared to ESNs.

Spike-timing-dependent plasticity (STDP) is a type of synaptic plasticity well considered in neuroscience and biophysics [95–99]. It is a reasonable encapsulation of the phenomena observed in the brain that involves the modulation of synaptic weights on the basis of the relative spike times between pre and post-synaptic neurons. It has been argued that STDP rules have broad implications for learning and memory processes in the nervous system. In this work, STDP rules will be used to alter the synaptic weights in the reservoir layer in the presence of external stimu-

lation and empower the analysis of the change in entropy. One may also note that it will subsequently update the timing and frequency of the spike for each neuron in the reservoir, allowing for a rich repertoire of firing patterns both spatially and temporally.

# Chapter 2

# Modelling advanced properties of associative learning in genetic circuits

## 2.1 Introduction

Synthetic biology is an emerging field that involves re-engineering existing biological systems or creating new ones that may solve real-world problems in medicine, agriculture, etc [100–108]. Over the past few decades, synthetic biology has witnessed a rapid revolution in the biotechnology industry, and opened up enormous potential for next-generation research in biology due to the increasingly tremendous power of genetic engineering technology, and ever decreasing cost of synthesis and sequencing [7, 109–115]. Particularly, increasing attention has been drawn to designing and testing synthetic biological circuits in vitro, in vivo and in silico in an attempt to better understand bioartificial intelligence at the cellular and molecular levels [116–121]. These artificial circuits can, therefore, function as fundamental units to modify existing cellular behaviours, and to perform a wide range of tasks of our own interest in programmable organisms [122–126]. Numerous synthetic circuits were developed for associative learning, decision making, and oscillators, and a brief summary is exhibited in Table 2.1 [30–35, 127–129]. With the growing collaboration between theorists and experimentalists in almost every discipline, we also noticed a trend in synthetic biology that mathematical models are frequently used to acquire insight, inform troubleshooting, and perform predictions [130–133].

Associative learning occurs in many aspects of our life, and it is reckoned to form the basis of our understanding of other forms of behaviours and cognition in human and nonhuman animals [134–138]. The most classical experiment on associative learning is the Pavlov's dog, in which the dog associated the ring of a bell with the smell of food [139,140]. The dog learned to associate the conditioned stimulus (bell's ring) with the unconditioned stimulus (smell), such that next time, in the presence of the bell's ring alone, the dog knew that the food would be served soon, and the learned response (the saliva from its mouth) was observed. The historical viewpoint is that the mammalian nervous system plays a vital role in associative learning through neuronal signaling and reconfiguration [141–144].

**Table 2.1:** Types of synthetic circuits.

| **Associative Learning** | **Decision Making** | **Oscillator** |
|---|---|---|
| Fernando et al., 2009 [40] | Nene et al., 2012 [31] | Stricker et al., 2008 [128] |
| Nesbeth et al., 2016 [116] | Filicheva et al., 2016 [32] | Tigges et al., 2009 [129] |
| Macia et al., 2017 [35] | Abrego and Zaikin, 2017 [33] | Borg et al., 2022 [127] |

However, some studies revealed the possibility that non-neural agents may also organise in a similar fashion [35, 145, 146]. Naturally, molecular circuits may display similar behaviours as molecular reactions form the building block of cellular activities. As a result, the design and investigation of molecular interactions that manifest associative learning have become active research topics in recent years. Although logic gates have been widely adopted in synthetic biology for emulating diverse biological behaviours [147–151], in this chapter, the focus is confined to the continuous models constructed by the Hill equations [152, 153], since continuous models tend to generate more accurate results and facilitate the comprehension of fine details of the system.

One of the first models along the line is the Fernando's model [40]. The work stands as a well-organised interdisciplinary article in which mathematical modelling and the bioengineering approach were given equal coverage and attention. In this work, the authors first built a genetic circuit motivated by the Hebbian learning which is reckoned prevalent in neural networks. A system of ordinary differential equations (ODEs) was later constructed using the Hill equation and the response

levels were visualised when subject to different conditions in order to exhibit the occurrence of associative learning. The article concluded with a brief description of the practical implementation in a single cell. Nonetheless, partly due to its interdisciplinary nature, the work only demonstrated that learning can be formed after conditioning with a brief simulation study and an in-depth analysis of the dynamics was still lacking. Furthermore, it may be worthwhile to consider some advanced properties beyond the formation of learning itself so as to increase the scope of applications in practical scenarios.

More specifically, it is natural to assume that the learned response becomes stronger with increasing times of conditioning. For illustrative brevity, we name the phenomenon *reinforcement effect*. It is also reasonable to suppose that the response becomes weaker with the repeated cuing of the conditioned stimulus (with no unconditioned stimulus taking place at the same time) shortly after the formation of conditioning, and we call it *forced dissociation*. To the best of our knowledge, neither of them has been formally discussed in the previous literature. The potential importance of these two behaviours can be explained from two aspects. On the one hand, in the Pavlov's dog experiment, the former would be equivalent to the scenario that the repeated conditioning of the bell and the food would reinforce the dog's belief that the bell's ring is a reminder of food availability; the latter would be that the repeated bell's ring alone shortly after the conditioning would stop the dog reckoning that the bell is related to the food's availability. On the other hand, these properties may provide experimentalists with more flexibility over the control of some biological systems, as synthetic circuits are widely used to regulate them [103, 105, 154, 155]. We show that the Fernando's model is able to manifest the reinforcement effect by choosing the proper parameters, but is unable to manifest forced dissociation. This motivated us to design a new circuit that possesses the potential to display forced dissociation. The new circuit also involves fewer proteins and does not contain any feedback loop, which can potentially reduce the wiring complexity in practical implementation. In the meantime, we also study the robustness of the respective models to the Hill coefficients, as this instructs experi-

mentalists on the type of polymers that can be used to implement the circuit.

## 2.2 Models

### 2.2.1 Fernando's Model

The circuit diagram of the Fernando's model is shown in Figure 2.1, and we assume that the circuit can be implemented in a programmable cell. Unlike the schematic

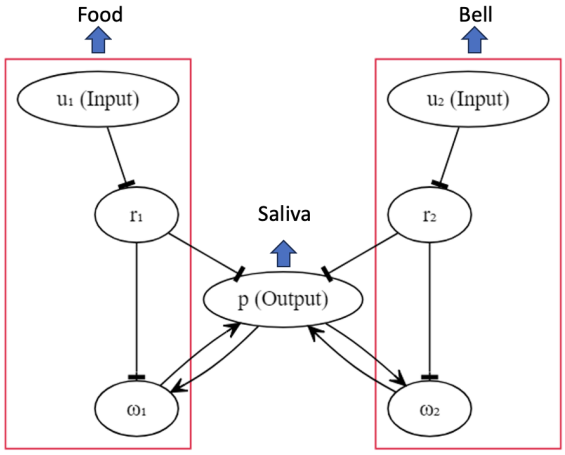



**Figure 2.1:** The schematic circuit of the Fernando's model. The correspondence to the components in the Pavlov's dog (indicated by blue arrows) is included for ease of reading.

diagram given in [40], here we omit genes for illustrative simplicity. In the diagram, each oval box denotes a particular protein. The activation is drawn with an arrow, and the inhibition is drawn with a hammerhead. Except for inhibitions $u_1 \to r_1$ and $u_2 \to r_2$, where the input molecules are directly bound to the repressors, all other activations and inhibitions are realised by transcription and translation. For instance, repressor $r_1$ inhibits the transcription of a particular gene, which guides the manufacture of molecule $\omega_1$. The upstream gene of protein $p$ has two available operator sites, one for $r_1$, $\omega_1$ and another for $r_2$, $\omega_2$. A more detailed explanation can be found in [40]. The Fernando's model is characterised by the system as follows

($N = 2$):

$$\begin{cases} \dfrac{dp}{dt} = \displaystyle\sum_{j=1}^{N} v_p\Big(\frac{\omega_j^a}{K_\omega^a + \omega_j^a}\Big)\Big(\frac{K_r^b}{K_r^b + r_j^b}\Big) - \delta_p p \\ \dfrac{d\omega_j}{dt} = v_\omega\Big(\dfrac{p^b}{K_p^b + p^b}\Big)\Big(\dfrac{K_r^b}{K_r^b + r_j^b}\Big) - \delta_\omega \omega_j + \varepsilon_j \\ r_j = \dfrac{R}{1 + k u_j} \end{cases} \tag{2.1}$$

In the equations above, $u_1$ and $u_2$ represent the respective concentrations of the unconditioned and conditioned stimuli, and they are given to the cell in a transient time window. $\omega_1$ and $\omega_2$ represent the respective concentrations of the weight molecules. $r_1$ and $r_2$ represent the respective concentrations of the repressor molecules. The concentration of the response molecule is denoted by $p$. $K_\omega$, $K_r$ and $K_p$ denote the respective Hill constants for molecule $\omega$, $r$ and $p$ that measure the concentrations of the transcription factors required for half occupancy. $R$ denotes the repressor concentration in the absence of molecule $u$. $a$ and $b$ are Hill coefficients that measure the cooperativity of the respective transcription factors. In [40], the authors used $a = 4$ and $b = 2$; however, in this work, we will also study the impact of varying integer values of $a$ and $b$ on qualitative behaviours. $\varepsilon$ denotes the basal grow rate, and we assume that it is only nonzero for $j = 1$. $v$ and $\delta$ denote the growth and degradation rate parameter, respectively, and the subscripts are used to signify the source of contribution. The architecture shows that the genetic circuit is structurally symmetric, and the left and the right halves are independent. The association is triggered by the feedback of the response molecule $p$, and the inspiration is derived from the Hebbian learning that dictates information exchange between neurons [156].

In order for an association to be formed, we can simply render the concentration of the molecule $\omega_1$ abundant, and the molecule $\omega_2$ insignificant before the start of the experiment. When only molecule $u_1$ is given to the cell, it is bound to the repressor molecule $r_1$ and reduces the concentration of molecule $r_1$. Therefore, the inhibition of the transcription with respect to the genes controlled by molecule $r_1$ is lifted. Eventually, sufficient molecule $\omega_1$ activates the transcription of the gene

associated with the response molecule $p$ and promotes the production of $p$. Conversely, when only molecule $u_2$ is given to the cell, we will not be able to observe abundant molecule $p$ due to the shortage of $\omega_2$ availability. However, at the time when molecule $u_1$ is paired with molecule $u_2$, the production of molecule $p$ (triggered by $u_1$) elevates the concentration of molecule $\omega_2$ because of the feedback loop, so that the next time, even when only molecule $u_2$ is present, there will already exist sufficient molecule $\omega_2$ for the production of $p$, which implies that the association has been formed.

In order to analyse a system of differential equations, one often converts the system to a dimensionless scale as a first step. One generally reduces the volume of parameters and removes physical units from the system, which facilitates mathematical investigations and renders the model more flexible for experimentalists who wish to implement the system in vivo or in vitro, as the units are not specified. By using the scaling $\overline{\omega_j} = \frac{\omega_j}{K_\omega}$, $\overline{r_j} = \frac{r_j}{K_r}$, $\overline{p} = \frac{p}{K_p}$, $\bar{t} = \delta_p t$, $\overline{u_j} = k u_j$, the dimensionless model becomes (overlines were dropped for simplicity):

$$\begin{cases} \dfrac{dp}{dt} = \displaystyle\sum_{j=1}^{N} \alpha \Big( \frac{\omega_j^a}{1+\omega_j^a} \Big) \Big( \frac{1}{1+r_j^b} \Big) - p \\ \dfrac{d\omega_j}{dt} = \beta \Big( \dfrac{p^b}{1+p^b} \Big) \Big( \dfrac{1}{1+r_j^b} \Big) - \theta \omega_j + \tau_j \\ r_j = \dfrac{S}{1+u_j} \end{cases} \tag{2.2}$$

where $\alpha = \frac{v_p}{K_p \delta_p}$, $\beta = \frac{v_\omega}{K_\omega \delta_p}$, $\theta = \frac{\delta_\omega}{\delta_p}$, $\tau_j = \frac{\varepsilon_j}{K_\omega \delta_p}$, $S = \frac{R}{K_r}$.

We can now try to intuitively interpret whether the reinforcement effect can be realised by the Fernando's model or not. Suppose we carry out the conditioning twice in order to guarantee that the second learned response is more abundant than the first, the simplest way is to ensure that the initial concentration of $\omega_2$ is small, and the growth parameter $\beta$ is not too large, so that we could anticipate $p$ to keep growing with the repeated conditioning according to Equation (2.2). Additionally, we need to select a relatively small $\theta$ to guarantee that the memory that is reflected by the weight molecules will not disappear too rapidly. Next, we investigate how the

two stimuli dissociate. Intuitively, in light of the design of the Fernando's model, the disappearance of the learned response is dictated by the time elapse. This is due to the fact that $\omega_2$ promotes the production of $p$, such that the response eventually disappears only if $\omega_2$ falls to 0. Additionally, the learned response does not attenuate if the time interval between the two successive stimuli is insufficient relative to the decay rate. In all, the dissociation is autonomous and is not dictated by the repeated cuing of the conditioned stimulus (alone). This can be circumvented by a different design, which is introduced next.

### 2.2.2 A Model with Forced Dissociation

The circuit for the model proposed here was inspired by [116] and is shown in Figure 2.2. In the diagram, each oval box denotes a particular protein, and we

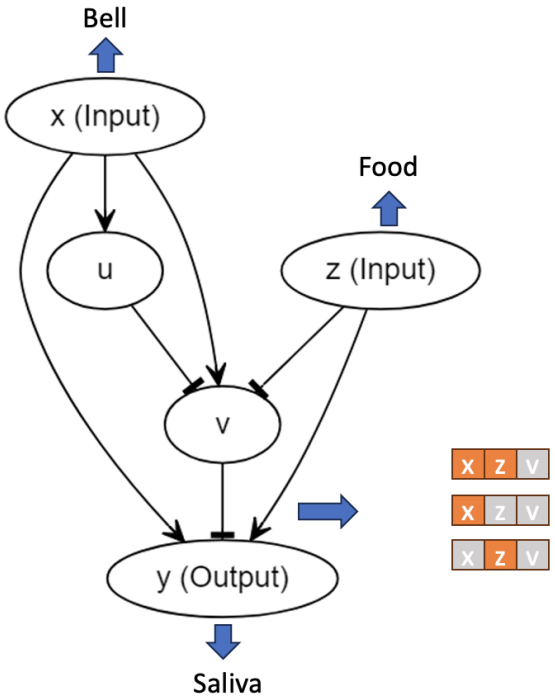



**Figure 2.2:** The schematic circuit of the model with forced dissociation. The correspondence to the components in the Pavlov's dog (indicated by blue arrows) is included for ease of reading. Also for the sake of clarity, the wiring that results in the production of the molecule $y$ is also broken down in order to detail different binding scenarios that contribute to the initiation of translation and transcription. The orange square indicates the presence of the molecule on the operate site whereas the grey square indicates the absence.

again omit the genes for the illustrative brevity. More specifically, input $x$ initiates

the transcription of a particular gene which guides the manufacture of molecule $u$; the translation of molecule $v$ is controlled by another gene, the expression of which is dictated by $x$, $u$ and $z$ together. Similarly, $y$ is controlled by a third gene, the expression of which is dictated by $x$, $v$ and $z$. The activation is drawn with an arrow, and the inhibition is drawn with a hammerhead. Input $x$ plays the role of the conditioned stimulus, input $z$ plays the role of the unconditioned stimulus, and $y$ represents the response. However, the discrimination between the conditioned and unconditioned stimuli has been downplayed in some sense; we will explain this in the discussion part. The design of this new architecture comes with at least two purposes. First, we intend to construct a simple circuit of which the mechanism is completely different from that of the Fernando's model, aiming for the components of the circuit to be more interactive with each other, and the association not to be reliant upon the participation of the feedback loops. These may lift a few restrictions on the synthetic implementation. Second, given that the Fernando's model is not capable of demonstrating forced dissociation, we hope to build a model that could successfully dissociate the two stimuli by repeating the cuing of the conditioned stimulus alone right after the conditioning. To fulfil the latter requirement, instead of placing a molecule ($\omega_2$ in Fernando's model) that promotes the transcription of the response protein molecule, we could actually consider introducing an inhibitor upstream of the response molecule. In this way, the consistent input of conditioned stimulus alone is expected to reduce the amount of the response molecule, so long as the stimulus promotes the expression of the inhibitor after the conditioning of the conditioned and unconditioned stimuli. This explains why we introduce the $x \to v \to y$ pathway in Figure 2.2. Regarding the other parts of the circuit, $z \to y$ and $z \to v \to y$ guarantee that input $z$ can always activate the output $y$, $x \to y$ ensures that there can exist a sufficient learned response upon the formation of associative learning, and $u$ functions as a moderator, speeding up the consumption of $v$ that renders the reinforcement effect more likely to occur. Here, we assume that the Hill coefficients for all molecules are identical, and they are denoted by $a$.

By employing the Hill equation, the system can be described with the following

equations:

$$\begin{cases} \dfrac{dy}{dt} = \alpha_{yx}\Big(\dfrac{x^a}{K_x^a + x^a}\Big)\Big(\dfrac{K_z^a}{K_z^a + z^a}\Big)\Big(\dfrac{K_v^a}{K_v^a + v^a}\Big) \\ \qquad + \alpha_{yz}\Big(\dfrac{K_x^a}{K_x^a + x^a}\Big)\Big(\dfrac{z^a}{K_z^a + z^a}\Big)\Big(\dfrac{K_v^a}{K_v^a + v^a}\Big) \\ \qquad + \alpha_{xyz}\Big(\dfrac{x^a}{K_x^a + x^a}\Big)\Big(\dfrac{z^a}{K_z^a + z^a}\Big)\Big(\dfrac{K_v^a}{K_v^a + v^a}\Big) \\ \qquad - \delta_y y \\ \dfrac{du}{dt} = \alpha_{ux}\dfrac{x^a}{K_x^a + x^a} - \delta_u u \\ \dfrac{dv}{dt} = \alpha_{vx}\Big(\dfrac{x^a}{K_x^a + x^a}\Big)\Big(\dfrac{K_u^a}{K_u^a + u^a}\Big)\Big(\dfrac{K_z^a}{K_z^a + z^a}\Big) - \delta_v v \end{cases} \tag{2.3}$$

Similarly to the Fernando's model, $K_x$, $K_z$, $K_v$, and $K_u$ denote the Hill constants for molecules $x$, $z$, $v$, and $u$, respectively. The production and degradation rates are denoted by $\alpha$ and $\delta$, where the subscripts are used to signify the source of contribution. Then, we use a similar scaling approach to render the system dimensionless and the dimensionless system is shown below:

$$\begin{cases} \dfrac{dy}{dt} = \alpha_{yx}\Big(\dfrac{x^a}{1 + x^a}\Big)\Big(\dfrac{1}{1 + z^a}\Big)\Big(\dfrac{1}{1 + v^a}\Big) \\ \qquad + \alpha_{yz}\Big(\dfrac{1}{1 + x^a}\Big)\Big(\dfrac{z^a}{1 + z^a}\Big)\Big(\dfrac{1}{1 + v^a}\Big) \\ \qquad + \alpha_{xyz}\Big(\dfrac{x^a}{1 + x^a}\Big)\Big(\dfrac{z^a}{1 + z^a}\Big)\Big(\dfrac{1}{1 + v^a}\Big) \\ \qquad - y \\ \dfrac{du}{dt} = \alpha_{ux}\dfrac{x^a}{1 + x^a} - \beta_u u \\ \dfrac{dv}{dt} = \alpha_{vx}\Big(\dfrac{x^a}{1 + x^a}\Big)\Big(\dfrac{1}{1 + u^a}\Big)\Big(\dfrac{1}{1 + z^a}\Big) - \beta_v v \end{cases} \tag{2.4}$$

One of the drawbacks of this circuit is that the model is not as heuristic as the Fernando's model. Indeed, the Fernando's model borrows the architecture of Hebbian learning, whereas we built our model from scratch, tailored for the properties that we wanted to achieve. Intuitively, in order for the reinforcement effect to occur, we anticipate that, by properly choosing the parameter values, the presence of $x$ alone could significantly elevate the amount of $v$, and the conditioning (whenever both $x$ and $z$ are present) could speed up the consumption of $v$. This ensures

that the second learned response is more abundant than the first, and the response to the conditioned stimulus alone is simultaneously less abundant than that to the unconditioned stimulus. When it comes to forced dissociation, as we discussed previously, the repeated cuing of input $x$ alone after the conditioning could elevate the concentration of $v$, which subsequently reduces output $y$ as a result of $x \to v \to y$. The numerical result is given in the next section.

Before closing this section, we briefly mention a potential adjustment to the current circuit that could simplify the system given in Equation (2.4). Our existing scheme allows for $x$, $v$, and $z$ to bind at a single operator site. In fact, we can adjust the output part in a way such that $z$ exploits the operator site that only restricts to $z$ itself, which makes it look somehow analogous to the output part of the Fernando's model, in which the left half and the right half are unrelated. The dimensionless model arising from the adjustment can then be reduced to:

$$\begin{cases} \dfrac{dy}{dt} = \alpha_{yx}\Big(\dfrac{x^a}{1+x^a}\Big)\Big(\dfrac{1}{1+v^a}\Big) + \alpha_{yz}\Big(\dfrac{z^a}{1+z^a}\Big) - y \\ \dfrac{du}{dt} = \alpha_{ux}\dfrac{x^a}{1+x^a} - \beta_u u \\ \dfrac{dv}{dt} = \alpha_{vx}\Big(\dfrac{x^a}{1+x^a}\Big)\Big(\dfrac{1}{1+u^a}\Big)\Big(\dfrac{1}{1+z^a}\Big) - \beta_v v \end{cases} \tag{2.5}$$

The adjustment provides an alternative with a simpler mathematical formulation (but likely with more biological complexity) for readers who wish to implement our circuit. We will demonstrate later on that this adjusted model can also display the same qualitative behaviours.

## 2.3 Results

Figure 2.3a displays one simulation result for the Fernando's model (Equation (2.2)), and the parameter values used in the simulation are listed in Table 2.2. Here, we used the Hill coefficients recommended in [40], which are $a = 4$ and $b = 2$.

The first spike in $p$ was stimulated by the unconditioned stimulus $u_1$. The second (small) spike in response $p$ was triggered by the conditioned stimulus $u_2$. Of course, the response $p$ could be adjusted to 0 only in the presence of $u_2$ by setting the initial concentration of $\omega_2$ to 0 just as in the original paper. Here, we highlight

the fact that various levels of the conditioned response are available to be chosen. The first conditioning was formed at the third spike, and the second conditioning was formed at the fifth spike in $p$ when both $u_1$ and $u_2$ were present. The fourth and the sixth spikes in $p$ represent the first and the second learned responses, respectively, when solely $u_2$ is present. The learned response was reinforced after repeated conditioning. Next, we move on to the discussion of how the two stimuli dissociate, namely, how the learned response is attenuated in the presence of consecutive conditioned stimuli alone after the formation of conditioning. The conjecture in the previous section was validated by the last four spikes in response $p$ in Figure 2.3a. As is apparent, the sixth, seventh, and eighth spikes were of the same amplitude, as the time intervals were not wide enough. Conversely, the response started to decrease (shown by the last two spikes) when the time interval was further widened. This could be deemed to be a limitation for the model because, in some applications (e.g., immune inflammation), we may hope to force the stimuli to dissociate by the repeated cuing of the conditioned stimulus alone in a short time window after the formation of associative learning, therefore establishing one of the motivations for our novel design.

Furthermore, we studied whether the qualitative behaviours of associative learning that we previously introduced are preserved or not apart from using the Hill coefficients recommended in [40]. The values for the other parameters remain the same, as shown in Table 2.2.

First, we fixed $b = 2$ and altered the Hill coefficient for the weight molecules from $a = 1$ to $a = 4$. The respective responses of molecule $p$ are displayed in Figure 2.4. As is obvious from the figure, the qualitative behaviours barely changed irrespective of the value of $a$, apart from the fact that $a = 1$ gave rise to a relatively notable response when only conditioned stimulus $u_2$ was present prior to conditioning. However, the result of $a = 1$ could still be classified as a valid associative learning in broad terms, as the response triggered by the conditioned stimulus $u_2$ alone is more significant after conditioning than before.

Then, we studied the case when $a = b$, and we altered $a$ from $a = 1$ to $a = 4$.

The respective responses of molecule $p$ are displayed in Figure 2.5, which shows that only $a = 1$ gave rise to undesirable behaviours, as the concentration of response $p$ never came down to 0. This is because $a = 1$ leads to a large transient growth rate of the weight for conditioned stimulus $\omega_2$.

In all, the Fernando's model is robust to the variation in Hill coefficients even without exploring the other parameters. It may offer more flexibility to synthetic biologists, since two dimers are not required to be bound cooperatively for weight molecules $\omega_1$ and $\omega_2$. As we show in Figure 2.4, even $a = 1$ and $b = 2$ could produce desirable results, which may reduce the experimental complexity.

**Table 2.2:** Parameter values used in the simulation for the Fernando's model (Equation (2.2)).

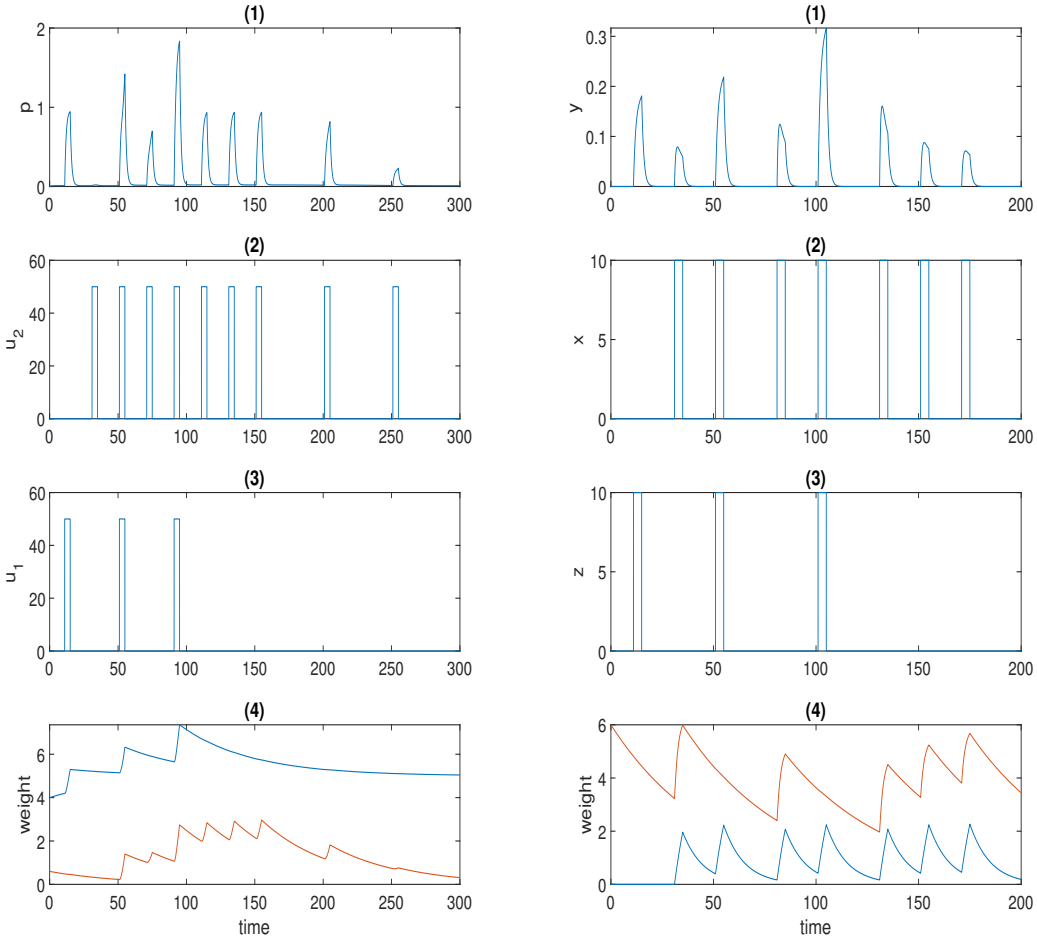
| Parameter | Value |
|---|---|
| $\alpha$ | 1 |
| $\beta$ | 0.8 |
| $\theta$ | 0.02 |
| $\tau$ | 0.1 |
| $S$ | 10 |

Figure 2.3b displays the simulation result for the model with forced dissociation 2.4 using $a = 2$, and the other parameter values used in the simulation are listed in Table 2.3.

**Table 2.3:** Parameter values used in the simulation for the model with forced dissociation (Equation (2.4)).

| Parameter | Value |
|---|---|
| $\alpha_{yx}$ | 2 |
| $\alpha_{yz}$ | 4 |
| $\alpha_{xyz}$ | 4 |
| $\alpha_{ux}$ | 0.6 |
| $\alpha_{vx}$ | 1.5 |
| $\beta_u$ | 0.1 |
| $\beta_v$ | 0.02 |

The first spike in response $y$ was stimulated by unconditioned stimulus $z$, and the second spike was stimulated by conditioned stimulus $x$. The first conditioning was formed at the third spike, and the second conditioning was formed at the fifth




**(a)** **(b)**

**Figure 2.3:** (**a**) Time series for the Fernando's model: (1) response molecule $p$; (2), (3) conditioned stimulus $u_2$ and unconditioned stimulus $u_1$; (4) weight molecule $\omega_1$ (blue) and $\omega_2$ (red); $\alpha = 1, \beta = 0.8$, $\theta = 0.02, \tau = 0.1, S = 10$. (**b**) Time series for the model with forced dissociation (1) response molecule $y$; (2), (3) conditioned stimulus $x$ and unconditioned stimulus $z$; (4) weight molecule $u$ (blue) and $v$ (red); $a = 2, \alpha_{yx} = 2, \alpha_{yz} = 4, \alpha_{xyz} = 4, \alpha_{ux} = 0.6, \alpha_{vx} = 1.5, \beta_u = 0.1, \beta_v = 0.02$. In (1), the first spike was stimulated by the unconditioned stimulus, and the second spike was triggered by the conditioned stimulus; the first and second conditionings were formed at the third and fifth spikes, respectively; the first and the second learned responses were reflected by the fourth and the sixth spikes; the remaining spikes demonstrate whether the forced dissociation could be realised or not.

spike in $p$ when both $z$ and $x$ were present. The fourth and sixth spikes in $p$ represent the first and the second learned responses, respectively. The learned response was

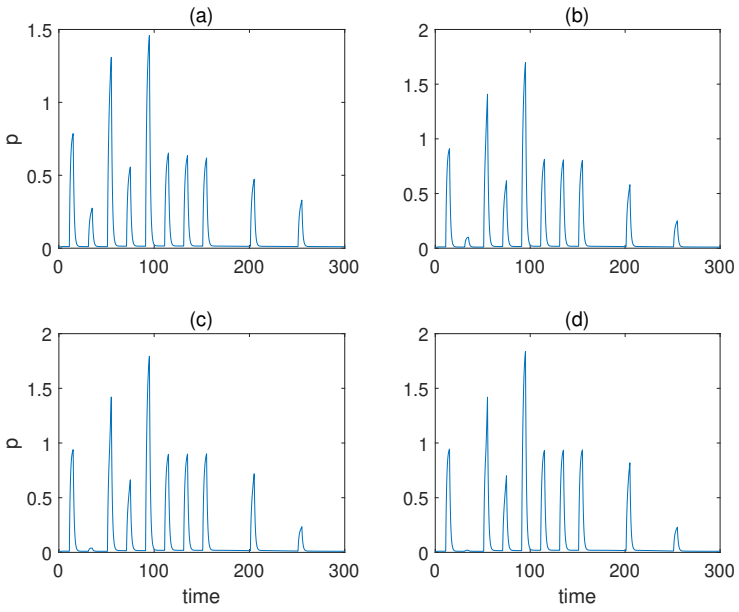



**Figure 2.4:** The concentration of molecule $p$ under various Hill coefficients $a$ when the value of $b$ is fixed for the Fernando's model. (**a**–**d**) Results when $a = 1, 2, 3, 4$. $b = 2$, $\alpha = 1$, $\beta = 0.8, \theta = 0.02, \tau = 0.1, S = 10$, respectively.

reinforced after repeated conditioning. As opposed to the Fernando's model, this model could successfully repress the learned response to the preconditioned level by means of the repeated cuing of the conditioned stimulus within a short time window, which was corroborated by the last two spikes in response $y$ in Figure 2.3b.

Considering that our model is not as heuristic as the Fernando's model, it is necessary to validate that the reinforcement effect is indeed the result of the conditioning. Therefore, we removed the second and the third unconditioned stimuli $z$ from the system, and the result is shown in Figure 2.6a . As can be seen, the reinforcement effect no longer existed without the conditioning of $x$ and $z$. The result also validates our conjecture in the previous section that the presence of $x$ and $z$ speeds up the degradation of $u$, which is a game changer for the formation of the reinforcement effect.

Similarly to what we performed for the Fernando's model, we studied the behaviours of this model under various Hill coefficients $a$ while keeping the other parameters specified in Table 2.3 unchanged.

From $a = 1$ to $a = 4$, the respective responses of molecule $y$ are displayed in

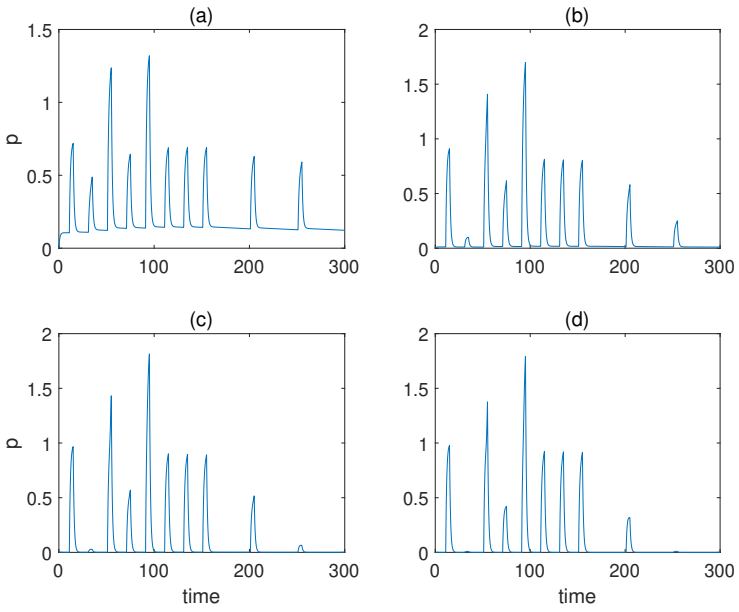



**Figure 2.5:** The concentration of molecule $p$ under various Hill coefficients when $a = b$ for the Fernando's model. (**a**–**d**) Results when $a = 1, 2, 3, 4$. $\alpha = 1, \beta = 0.8, \theta = 0.02$, $\tau = 0.1, S = 10$, respectively.

Figure 2.7. As can be observed, only $a = 2$ and $a = 3$ yielded desirable associative learning behaviours. The effect of conditioning is not discriminative for $a = 1$, and for $a = 4$, the response during conditioning is not more significant than the one when only the unconditioned stimulus is present.

The previously introduced adjusted model (Equation (2.5)) could also display the qualitative behaviours that are shown in Figure 2.6b when using $a = 2$. The parameter values used in this simulation are listed in Table 2.4. The adjusted model could give rise to a more abundant response on the dimensionless scale compared to the original version (Figures 2.3b and 2.6b).

Again, we studied the behaviours of the model with various Hill coefficients $a$ without changing the other parameters specified in Table 2.4. As is apparent from Figure 2.8, the qualitative behaviours of the associative learning are largely preserved. However, the learned responses for cases $a = 3$ and $a = 4$ are less significant than that of $a = 2$.

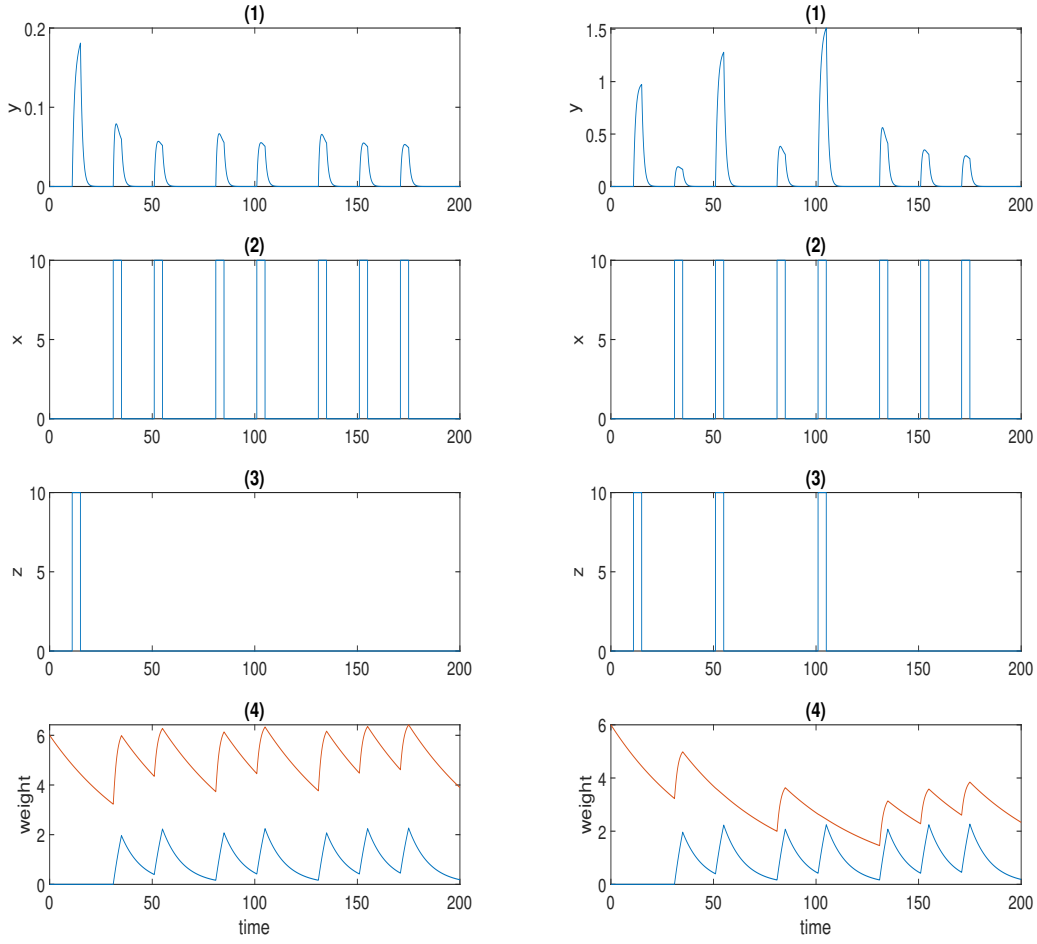



**Figure 2.6:** (**a**): Time series for the model with forced dissociation without conditioning (1) response molecule $y$; (2), (3) conditioned stimulus $x$ and unconditioned stimulus $z$; (4) weight molecule $u$ (blue) and $v$ (red); $a = 2, \alpha_{yx} = 2, \alpha_{yz} = 4, \alpha_{xyz} = 4, \alpha_{ux} = 0.6, \alpha_{vx} = 1.5, \beta_u = 0.1, \beta_v = 0.02$. (**b**) Time series for the adjusted model with forced dissociation (1) response molecule $y$; (2), (3) conditioned stimulus $x$ and unconditioned stimulus $z$; (4) weight molecule $u$ (blue) and $v$ (red); $a = 2, \alpha_{yx} = 4, \alpha_{yz} = 1, \alpha_{ux} = 0.6, \alpha_{vx} = 1, \beta_u = 0.1, \beta_v = 0.02$.

## 2.4 Discussion

In this work, we presented a detailed analysis of two advanced behaviours (reinforcement effect and forced dissociation) in associative learning. The Fernando's model could successfully demonstrate the reinforcement effect if we properly chose the parameter values. However, the attenuation of the learned response only oc-

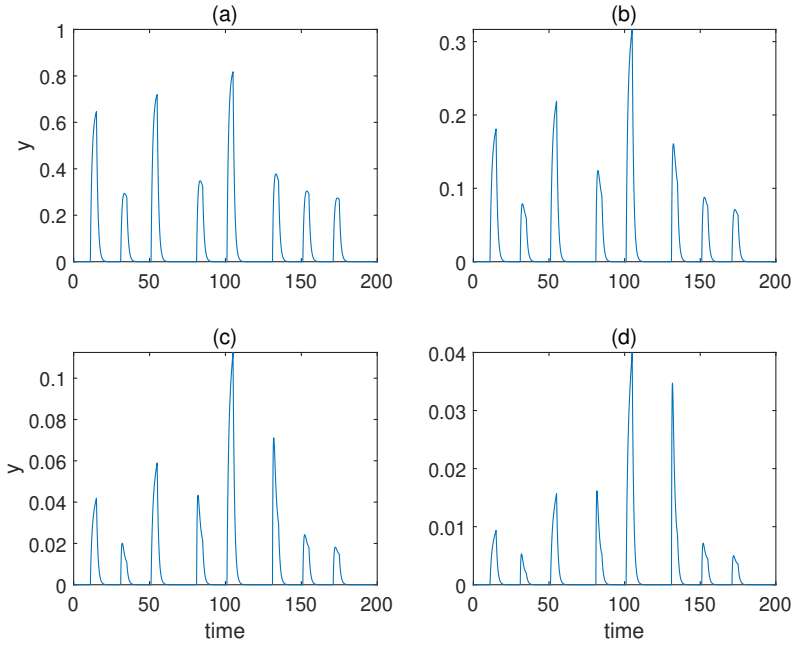



**Figure 2.7:** The concentration of molecule $y$ under various Hill coefficients $a$ for the model with forced dissociation. (**a–d**) Results when $a = 1,2,3,4$. $\alpha_{yx} = 2, \alpha_{yz} = 4, \alpha_{xyz} = 4$, $\alpha_{ux} = 0.6, \alpha_{vx} = 1.5, \beta_u = 0.1, \beta_v = 0.02$.

**Table 2.4:** Parameter values used in the simulation for the adjusted model with forced dissociation (Equation (2.5)).

| Parameter | Value |
|---|---|
| $\alpha_{yx}$ | 4 |
| $\alpha_{yz}$ | 1 |
| $\alpha_{ux}$ | 0.6 |
| $\alpha_{vx}$ | 1 |
| $\beta_u$ | 0.1 |
| $\beta_v$ | 0.02 |

curred when the time interval between the learned responses is large enough in the sense that there is no way to force the learned response to decrease within a short time window. The model introduced in Section 2.2.2 could manifest the reinforcement effect and forced dissociation with the parameter values listed in Table 2.3, which could potentially provide more possibilities for the biological and medical applications of synthetic biology.

Having highlighted the contribution of our model, we must point out that it comes with a few constraints of which synthetic biologists need to be aware. First,

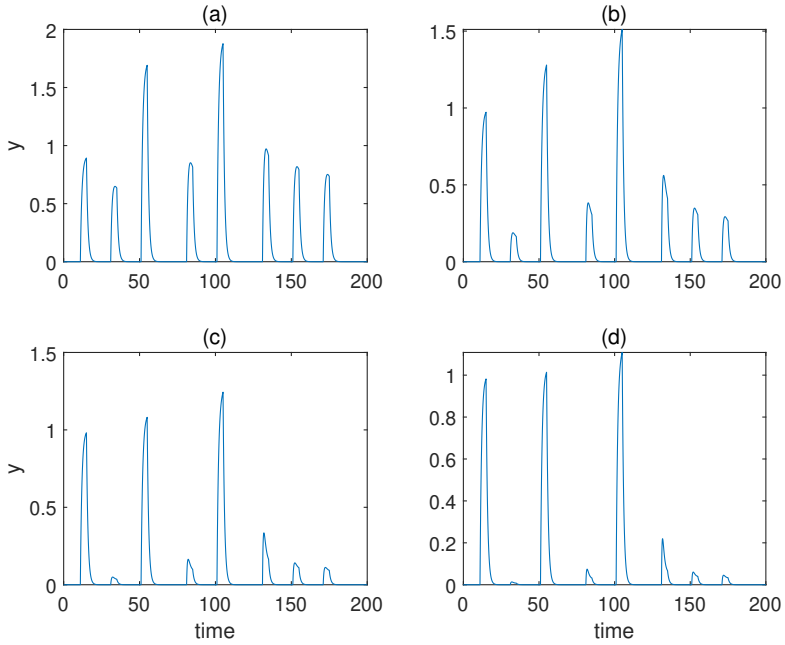



**Figure 2.8:** The concentration of molecule $y$ under various Hill coefficients $a$ for the adjusted model with forced dissociation. (**a**–**d**) Results when $a = 1,2,3,4$. $\alpha_{yx} = 4, \alpha_{yz} = 1, \alpha_{ux} = 0.6, \alpha_{vx} = 1, \beta_u = 0.1, \beta_v = 0.02$.

the overall qualitative behaviour of the system is not very robust to the parameters aside from the Hill coefficients. We found that a 25% change in parameter values could lead to less desirable behaviours. An example is given in Figure 2.9, where $\alpha_{ux}$ was changed from 0.6 to 0.45. The first spike was then of the same amplitude as that of the third in response $y$. However, the stringent constraint can be relaxed if we expect to implement only one of the two behaviours, either reinforcement effect or forced dissociation. Second, it is extremely difficult, if not impossible, to control the response to the conditioned stimulus prior to conditioning at an insignificant level (second spike in $y$ in Figure 2.3b) while maintaining the behaviours of interest. Hence, the model proposed here may not be a suitable candidate to emulate the behaviours of the Pavlov's dog, but fits the context of associative learning in a broader sense where conditioning and learning are major concerns. Third, the model demands that $v$ remains abundant in order for associative learning to happen. Therefore, $v$ needs to be supplemented to a sufficient level before the start of each experiment. Otherwise, conditioned stimulus $x$ alone could result in an over-

expression of the response.

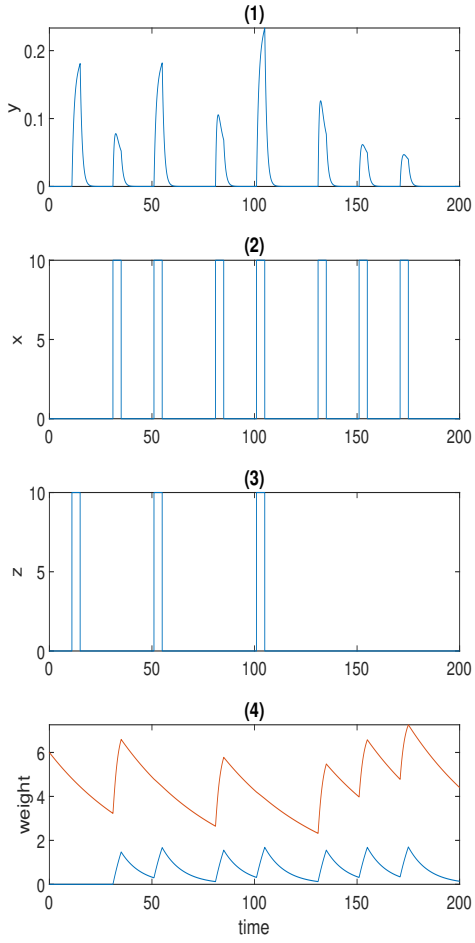



**Figure 2.9:** An example to illustrate the effect of change in parameter values on the model with forced dissociation. $a = 2, \alpha_{yx} = 2, \alpha_{yz} = 4, \alpha_{xyz} = 4, \alpha_{ux} = 0.45, \alpha_{vx} = 1.5, \beta_u = 0.1, \beta_v = 0.02$.

Last but not least, we also want to mention several potential applications of our work in the field of synthetic biology and medicine. First, in the treatment for diabetes, Ye et al. [157] built a synthetic signaling cascade that enhances blood-glucose homeostasis. The reinforcement effect that we demonstrated in the models may pave the way for further adjustment to the circuit in the hope to attain the more efficient control of glucose level. Second, in the treatment of immune-mediated diseases, adoptive T-cell transfer technology shows immense promise in the treatment of immune-mediated disease such as cancer immunotherapy [158]. The fea-

ture of forced dissociation displayed in our model may provide more flexibility for experimentalists to shut off an excessive immune response once the tumours are eliminated. Lastly, some recent studies were focused on modelling the network of neurodegenerative markers [159, 160]. The models that we discussed in this article may shed light on how to model these new findings at the genetic circuit level and build hierarchical neuronal architectures. The models may also play a supportive role in the existing technology that controls neurotransmitter release [161].

# Chapter 3

# Studying the impact of astrocytic coverage of synapses on the short-term memory

## 3.1 Introduction

Over the past few decades, the dynamics of neuronal networks have been widely studied [162–166] and increasing effort has been devoted to understanding the roles played by a type of glial cells, astrocytes [167–172]. Traditionally, astrocytes have been reckoned as auxiliary cells to neurons and it has now become evident that astrocytes can not only support the structure of the nervous system, but also modulate synaptic transmission [173–177]. Neuron–astrocyte coupling plays an indispensable role in the functioning of neuronal networks via bidirectional communication under the notion *tripartite synapse* [57–59, 178, 179]. It is found that astrocytes can sense the synaptic activities by the uptake of neurotransmitters released from the synaptic cleft and provide feedback to pre- and post-synaptic neurons via gliotransmitter release caused by the temporary elevation of intracellular calcium concentration which normally lasts seconds to minutes [180–184]. All these findings in molecular biology pave the way for a better understanding of the information processing in neuron–astrocyte circuits and the formation of cognitive functions. Very recently, mathematical and computational approaches have been used

to investigate the contribution of astrocytes to the organisation of spatial and temporal synchronization in neural networks [185–188], formation of working memory [63–65, 189, 190] and generation of integrated information in neuronal ensembles [191–194], which takes a step further to the understanding of the intelligence arising from the nervous system.

Nowadays, a widely accepted fact is that astrocytes play an active role in various types of memory and the memory improvement may be related to the change in the astrocyte density [195–199]. Working memory is the ability of an entity to retain limited information in a readily accessible form and provides an interface between memory and cognition [200–202]. Some biological evidence has already raised the possibility that astrocytes could be highly involved in working memory [203–206] and it has been well known that the astrocytic coverage of synapses is a highly dynamic process that alters throughout lifetime [207–209]. Therefore, it is natural to hypothesize that there exists a potential correlation between the working memory and the astrocytic coverage (or astrocyte density) but very little effort has been made so far. In this work, the astrocytic coverage is equivalent to the astrocyte density and it will become clear when we introduce our model. Furthermore, some studies have also revealed that the attenuation of calcium events correlates with the reduction of astrocytic coverage of asymmetric synapses in the hippocampal CA1 region in mice and that the size of the calcium events within astrocytes follows the power law [210–213]. Hence, this line of research may also help explain the cause of various sizes of calcium events.

Lately, an in silico neuron–astrocyte network model has been employed to manifest that astrocytes indeed assist the formation of short-term memory and mediate analogous memory [63–65, 189, 190]. It provides a quantitative score to measure the recall accuracy as a result of the short-term memory. In this work, we leverage this computational model and study the impact of varying astrocytic coverage areas of synapses on the short-term memory performance. Unlike in the original article [64], here, we focus on the performance of the single-item task so as to ensure that the real pattern remains unchanged throughout the experiment. We also intro-

duced a low-pass filter to the input image in order to alter the spatial frequencies. In particular, it is of great interest to learn how the change in the number of spatial frequency components will impact the short-term memory and how the relation is affected by the astrocytic coverage. The input image is also subject to different levels of the salt-and-pepper noise to make our evaluation more comprehensive. We demonstrate that the short-term memory performance is significantly affected by the astrocytic coverage. Additionally, we also underlie some other observations that may interest biologists.

## 3.2 Models and Methods

Our work employed the neuron–astrocyte network developed in [64] and an illustrative diagram for the architecture is shown in Figure 3.1. From left to right are the input image, neuronal network and astrocytic network, respectively. The neuronal network is of dimension $W \times W$ and the astrocytic network is of dimension $M \times M$. All neurons are excitatory and each astrocyte from the astrocytic network regulates an $l \times l$ neuronal square of the same size from the neuronal network. All neurons within the square are connected to the astrocyte. The connections in the neuronal network will be defined later and the astrocytes in the astrocytic layer are connected to their nearest neighbours vertically and horizontally. The input digital image is converted into electric current and is applied to the neuronal network in that one image pixel corresponds to exactly one neuron. The values used for the parameters introduced in Sections 3.2.1–3.2.3 are listed in [64], unless otherwise specified. The novel methodology used to study the impact of astrocytic coverage on short-term memory is introduced in Sections 3.2.4–3.2.6.

### 3.2.1 Neuronal Network

Considering that we were to simulate a relatively large network, we used the Izhikevich model to characterize the dynamics of neurons as it demonstrates computational efficiency while maintaining the biological properties of the canonical Hodgkin–Huxley model [214]. The comparison of the properties and the efficiency with the Hodgkin–Huxley model as well as a few other prevalent biophysical neu-

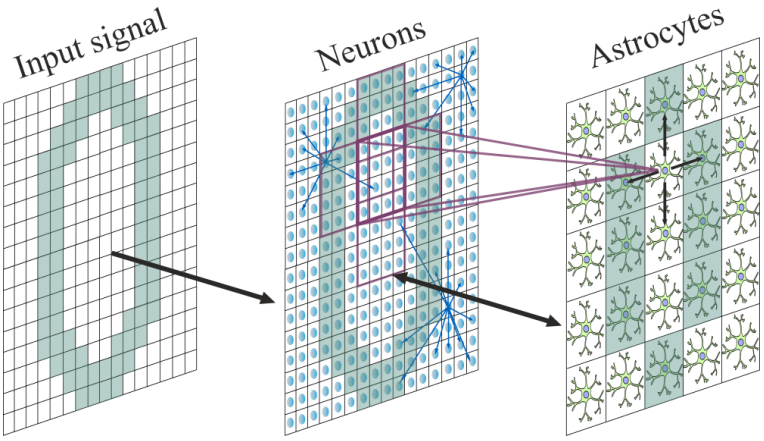



**Figure 3.1:** An illustrative diagram for the neuron–astrocyte network, courtesy of Yuliya Tsybina. Neurons receive the input signal converted from a digital pattern of the same size. The number of out-connections per neuron is fixed and the connections follow an exponential distribution. Each astrocyte regulates a neuronal square of the same size and neurons within the square are connected to the astrocyte. The adjacent neuronal squares have the same width of overlap. More details are in the main text.

ronal models is shown in Appendix B. The model is shown below:

$$\begin{cases} \dfrac{dV^{(i,j)}}{dt} = 0.04V^{(i,j)^2} + 5V^{(i,j)} - U^{(i,j)} + 140 + I_{app}^{(i,j)} + I_{syn}^{(i,j)} \\ \dfrac{dU^{(i,j)}}{dt} = a(bV^{(i,j)} - U^{(i,j)}) \end{cases} \tag{3.1}$$

where $V$ denotes the membrane potential of a particular neuron and $U$ represents the membrane recovery variable, with post-spike resetting: if $V^{(i,j)} \geq 30\text{mV}$, then

$$\begin{cases} V^{(i,j)} = c \\ U^{(i,j)} = U^{(i,j)} + d \end{cases} \tag{3.2}$$

The part $0.04V^{(i,j)^2} + 5V^{(i,j)} - U^{(i,j)} + 140$ was chosen to ensure the membrane potential $V$ has the $mV$ scale and the time $t$ has $ms$ scale. The model can also exhibit the known types of firing patterns of cortical neurons such as regular spiking, fast spiking, chattering, etc, in choosing the appropriate values of parameters $a$, $b$, $c$, $d$. The superscript $(i, j)$ denotes the positional index of the neuron. $I_{app}$ represents the applied input current converted from the digital image and $I_{syn}$ represents the

net current received from all presynaptic neurons which takes the form (generalized from [215,216]):

$$I_{syn}^{(i,j)} = \sum_k \frac{g_{syn}^{(i,j)}(E_{syn} - V^{(i,j)})}{1 + \exp(\frac{-V_{pre}^k}{k_{syn}})} \tag{3.3}$$

where the summation is over all presynaptic neurons. The synaptic weight is dictated by $g_{syn}^{(i,j)} = \eta + \nu^{(i,j)}$ where $\eta$ reflects the baseline weight and $\nu^{(i,j)}$ describes the impact of astrocytic calcium events which will be defined later. $E_{syn}$ denotes the reversal potential for excitatory synapses and $V_{pre}^k$ denotes the membrane potential of the neuron $k$. For clarity, we need to point out that the short-term synaptic plasticity is not considered in our model. We use $a = 0.1, b = 0.2, c = -65, d = 2$ in order to enable the occurrence of fast spiking in the neurons.

In this work, we fixed the number of out-connections per neuron as $N_{out}$ in that each presynaptic neuron interacts with $N_{out}$ postsynaptic neurons. The connections are established according to an exponential distribution with $R$ being the distance between each pair of neurons:

$$f(R) = \begin{cases} \frac{1}{\lambda}\exp(-R/\lambda) & R \geq 0 \\ 0 & R < 0 \end{cases} \tag{3.4}$$

### 3.2.2 Action Potential-Induced Elevation of Glutamate and $IP_3$

For each presynaptic neuron, the amount of glutamate, a type of neurotransmitter released into the synaptic cleft is dictated by the spiking events of the neuron [217]:

$$\frac{dG^{(i,j)}}{dt} = -\alpha_{glu} G^{(i,j)} + k_{glu}\Theta(V^{(i,j)} - 30) \tag{3.5}$$

where $\Theta$ denotes the Heaviside function.

$IP_3$ is a ligand and is produced in response to the external stimuli such as neurotransmitters [218]. It regulates many pathways including the release of $Ca^{2+}$ from Endoplasmic Reticulum (ER) into cytoplasm [219] which will be described in due course. The dynamics of the intracellular concentration of the molecule $IP_3$

within each astrocyte is described by

$$\frac{dIP_3^{(m,n)}}{dt} = \frac{IP_3^{\star} - IP_3^{(m,n)}}{\tau_{IP_3}} + J_{PLC_\delta}^{(m,n)} + J_{glu}^{(m,n)} + diff_{IP_3}^{(m,n)} \tag{3.6}$$

Here, $IP_3^{\star}$ denotes the steady state of the intracellular $IP_3$ concentration and $J_{PLC_\delta}$ encapsulates the $IP_3$ produced by phospholipase $C\delta$ which takes the form

$$J_{PLC_\delta} = \frac{v_4(Ca + (1-\alpha)k_4)}{Ca + k_4} \tag{3.7}$$

where $Ca$ denotes the $Ca^{2+}$ concentration in the astrocytic cytoplasm. We use $diff_{IP_3}$ to represent the diffusion of $IP_3$ via gap junctions between adjacent astrocytes and is given by

$$diff_{IP_3} = d_{IP_3}(\Delta IP_3) \tag{3.8}$$

where $\Delta IP_3$ denotes the discrete Laplace operator reflecting the diffusion as a result of $Ca^{2+}$ exchange with neighbouring astrocytes. The production of $IP_3$ stimulated by glutamate via metabotropic glutamate receptors (mGluRs) and phospholipase $C\beta$ is characterized by

$$J_{glu} = \begin{cases} A_{glu} & t_0 < t \leq t_0 + t_{glu} \\ 0 & \text{otherwise} \end{cases} \tag{3.9}$$

where $t_{glu}$ denotes the duration that persists since time $t_0$, when the total level of glutamate associated with a particular astrocyte reaches the threshold $F_{act}$:

$$\frac{1}{N_a} \sum_{(i,j) \in N_a} \Theta(G^{(i,j)} - G_{thr}) > F_{act} \tag{3.10}$$

Here, we use $t_{glu} = 0.06$s.

### 3.2.3 Astrocytic Network

Although voltage-gated calcium channels (VGCC) have been shown to be able to elevate intracellular calcium concentration and many authors included them in their models [220–223], here, we use the Ullah model [224] to simplify the description

of the calcium dynamics within astrocytes where only the impact of glutamate is considered:

$$\begin{cases} \dfrac{dCa^{(m,n)}}{dt} = J_{ER}^{(m,n)} - J_{pump}^{(m,n)} + J_{leak}^{(m,n)} + J_{in}^{(m,n)} - J_{out}^{(m,n)} + diff_{Ca}^{(m,n)} \\ \dfrac{dh^{(m,n)}}{dt} = a_2 \left( d_2 \dfrac{IP_3^{(m,n)} + d_1}{IP_3^{(m,n)} + d_3} (1 - h^{(m,n)}) - Ca^{(m,n)} h^{(m,n)} \right) \end{cases} \tag{3.11}$$

The explicit forms of the individual terms are summarized below:

$$\begin{cases} J_{ER} = c_1 v_1 Ca^3 h^3 IP_3^3 \dfrac{c_0/c_1 - (1+1/c_1)Ca}{(IP_3 + d_1)^3 (Ca + d_5)^3} \\ J_{pump} = \dfrac{v_3 Ca^2}{k_3^2 + Ca^2} \\ J_{leak} = c_1 v_2 (c_0/c_1 - (1+1/c_1)Ca) \\ J_{in} = \dfrac{v_6 IP_3^2}{k_2^2 + IP_3^2} \\ J_{out} = k_1 Ca \\ diff_{Ca} = d_{Ca}(\Delta Ca) \end{cases} \tag{3.12}$$

Here, $Ca$ denotes the $Ca^{2+}$ concentration within cytoplasm and $h$ denotes the fraction of opened $IP_3$ receptors (IP3Rs) on the ER. We assume that the astrocytes are spatially homogeneous. ER is a continuous membrane system that stores a reservoir of $Ca^{2+}$ within astrocytes. The released $IP_3$ then binds to IP3Rs on the ER and opens the channel allowing for the flow of $Ca^{2+}$ from the ER into the cytoplasm, which is characterized by $J_{ER}$. In this model, we assume the co-existence of the ER and the cytoplasm in individual astrocytes and the homogeneous distribution of ER in the interior of astrocytes. $J_{pump}$ denotes the ATP-dependent pump that recovers $Ca^{2+}$ from the cytoplasm back to the ER. $J_{leak}$ denotes the leakage of $Ca^{2+}$ from the ER to the cytosol due to the concentration gradient. $J_{in}$ and $J_{out}$ denote the $Ca^{2+}$ exchange with the extracellular space. $diff_{Ca}$ represents the diffusion of $Ca^{2+}$ via gap junctions. As one will see in Section 3.3, the diffusion may have a negative impact on short-term memory based on our toy example. However, the existence of propagation in the form of calcium signaling via gap junction has been verified

by a number of studies [225–227] for the cross-talk between astrocytes. Therefore, we retain the diffusion term in the model to emphasize this important biological property.

Finally, the calcium-dependent gliotransmitter-induced modulation of synaptic weight by the associated astrocyte via the N-methyl-D-asparate receptors (NMDARs) is defined as

$$v = v^{\star}\Theta(Ca - Ca_{thr}) \tag{3.13}$$

where $v^{\star}$ denotes the weight of the synapse as a result of the astrocytic modulation of synaptic transmission if the $Ca^{2+}$ concentration is beyond the threshold required for gliotransmitter release, $Ca_{thr}$, and the fraction of spiking neurons associated with that astrocyte during the time interval $\tau_{syn}$ is above $F_{astro}$. The duration of feedback is denoted by $\tau_{astro}$ and we use $\tau_{astro} = 250$ms.

### 3.2.4 Variation of Astrocytic Coverage

In order to study the short-term memory performance of the network under various astrocytic coverage areas, we need to vary the size of the astrocytic layer $M$. However, to ensure that each astrocyte modulates an identical size of neuronal square and there are no neurons left uncovered, the following equation must be satisfied:

$$\frac{W-1}{l-p} = M \tag{3.14}$$

where $p$ is the size of the overlapping edge and $l$ is the size of the neuronal square covered by each astrocyte. In this work, we fixed $p = 1$. Since the input image is of dimension $79 \times 79$, if $W = 79$, the equation is satisfied for $l = 2,3,4,7$. To analyze the effect of $l = 5,6,9$, the image is adjusted by adding a periphery of stripe of width 1 outside of the edge, the intensity of which is chosen to be the same as the background. Now $W = 81$ and the equation is satisfied. Similarly, the equation is satisfied for $l = 8$ by choosing $W = 78$ (edge removal on one side). In this way, the size of the image is by and large maintained and the digital patterns are least damaged.

### 3.2.5 Variation of Spatial Frequencies

In this work, we utilized low-pass filter to alter the spatial frequencies of the input image.

The 2D discrete forward Fourier transform converts the image from the spatial domain into the frequency domain with:

$$F(k,l) = \sum_{x=0}^{W-1}\sum_{y=0}^{W-1} f(x,y)e^{-i2\pi(\frac{kx}{W}+\frac{ly}{W})} \tag{3.15}$$

The inverse transform converts from the frequency domain back to the spatial domain with:

$$f(x,y) = \frac{1}{W^2}\sum_{k=0}^{W-1}\sum_{l=0}^{W-1} F(k,l)e^{i2\pi(\frac{kx}{W}+\frac{ly}{W})} \tag{3.16}$$

where $f(x,y)$ denotes the intensity at pixel $(x,y)$ whilst $F(k,l)$ consists of the spectrum and the phase angle at frequency $(k,l)$. In general, one is more concerned with the spectrum as compared to the angle so the angle is not within the scope of our discussion. Figure 3.2 displays what digit zero looks like in the spatial domain (left) and spectral domain (right), respectively. By convention, $F(0,0)$ is placed at the center of the spectral domain and is also the largest component of the image. Moreover, we display the frequency domain on the logarithmic scale so as to make the other frequency components more visible. The frequency increases as we move farther away from the center in the spectral domain.

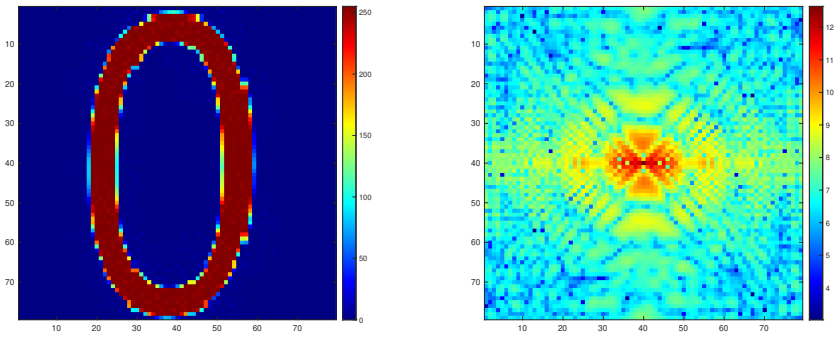


**Figure 3.2:** Spatial domain and spectral domain of digit 0.

A low-pass filter applies a threshold $f_0$ to the spectral domain and sets all the

components above $f_0$ to zero. In this work, the filter will be applied to the input current at each time step. By slowly increasing the threshold, we hope to figure out the sensitivity of the short-term memory performance to the change in spatial frequencies.

### 3.2.6 Simulation Protocols

Most of the parameter values and protocols used in this work are identical to those in [64]. Since the size of our parameters is huge, we do not list the values used in this article. One can refer to [64] for more details. Here, we only made a few adjustments in order to study the impact of spatial frequencies in a more effective way. Therefore, unless otherwise specified, one can assume that we herein use the same protocol as in [64].

The dynamics of the astrocytic network are simulated using the Runge-Kutta fourth-order method and the remaining part using the forward Euler method with time-step $\Delta t = 0.1$ ms. The input current $I_{app}$ is converted from a digital image (0–9) with the same size of the neuronal network by scaling the pixel intensity, which will be used in the learning and the testing stage. The pixel intensity is scaled in the range $[0, A_{stim}]$ for learning and $[0, A_{test}]$ for testing in order to prevent over-excitation of neurons. In this work, we employ the binary encoding that converts intensity over 127 to $A_{stim}$ ($A_{test}$) and to 0 otherwise. The input is also subject to salt-and-pepper noise which will also alter the frequency domain in addition to the low-pass filter. Different from a low-pass filter which will cut off the frequency components above a threshold, increasing the salt-and-pepper noise tends to include more frequency components (high frequency components in particular) as the noise will break the image down into pieces. We are interested in investigating the effect of both on the short-term memory performance. In this work, we do not introduce it in the learning stage so as to keep the real pattern intact and only alter the noise level in the testing stage. Here, our work is only focused on the single-item implementation. This is to ensure that the real spatial pattern is fixed during the experiment. Unless otherwise specified, in the learning stage, the input current $I_{app}$ is applied to the network at $t_1 = 0.1$ s for $t_{stim} = 200$ ms and in the testing stage, $I_{app}$ is applied

to the network at $t_2 = 2$ s for $t_{test} = 150$ ms. The simulation terminates at $t_e = 2.3$ s. We alter $t_2$ and $t_e$ when investigating the impact of the time interval between training and testing. In addition, note that changing the frequency domain will result in complex values in the spatial domain when conducting the inverse transform. To this end, we take the absolute values and re-scale them with $A_{stim}$ ($A_{test}$). Our simulation time is shorter as compared to the one used in [64] but the time interval between the learning and the testing stage is already long enough for the activation of calcium release within astrocytes. Besides, the time scale of the calcium events in astrocytes described by the calcium model in Section 3.2.3 is several seconds. It corresponds to the typical duration of short-term memory and the speed of calcium signaling in astrocytes [228].

### 3.2.7 Performance Measure

To measure the performance of our model, i.e., to what extent the model is able to memorize the real pattern, we came up with a correlation measure $C_p$ that compares the recalled pattern (during testing) with the real pattern:

$$\begin{cases} M_{ij}(t) = I\left[\left(\sum_{k=t}^{t+\omega} I[V_{ij}(k) > 30]\right) > thr\right] \\ CD(t) = \frac{1}{|P|}\sum_{(i,j)\in P} M_{ij}(t) \\ CB(t) = \frac{1}{W^2 - |P|}\sum_{(i,j)\notin P} (1 - M_{ij}(t)) \\ C(t) = \frac{1}{2}(CD(t) + CB(t)) \\ C_p = \max_{thr} C(t) \end{cases} \tag{3.17}$$

Here $t$ is the start time of the testing stage and we use $\omega = 250$ms. $P$ represents the set of pixels belonging to the real pattern. $CD$ represents the true positive rate in our context, namely, how many pixels that belong to the real pattern have been recalled. Similarly, $CB$ represents the true negative rate. Therefore, $C$ can accurately reflect the overall performance of the neuron–astrocyte network. We select $C_p$ that maximizes $C(t)$ over the whole-number thresholds, $thr = 1, 2, \ldots, 30$.

## 3.3 Results

In this section, we mainly show how short-term memory performance is affected by astrocytic coverage under various spatial frequencies and salt-and-pepper noise levels.

Figures 3.3 and 3.4 display the model's performance scores $C_p$ for digit zero (a symmetric digit) and two (an asymmetric digit) under various conditions, respectively. Each square represents one single simulation using the protocol described in the last section. In each sub-figure, the vertical axis denotes various levels of the salt-and-pepper noise at the testing stage. The horizontal axis denotes the moving threshold $f_0$ (increased by 2 units) of the low-pass filter from $f_0 = 4$ to $f_0 = 58$. Namely, more frequencies will be included as we move farther away from the origin. The plot starting from $f_0 = 4$ is to ensure the visual contrast for the performance over $f_0 = 4$ and we will explain in more detail why there exists a sharp rise in performance from 4 later. The filter threshold terminates at $f_0 = 58$ because it will already incorporate all frequency components with respect to the largest picture ($W = 81$) in this study.

On the one hand, for all sizes ($l = 2,3,4,5,6,7,8,9$) of the astrocytic coverage and noise levels during the testing stage, the performance plunges when moving the filter threshold from 5 to 4, which corresponds to $(5\times 2)/79 \approx 1/8 - (4\times 2)/79 \approx 1/10$ of the distance from the center to the edge in the spectral domain. On the other hand, the trace width of the digit in the image we use is about 8–10 pixels. By trace width we mean the interval between the boundaries of the digit. This makes it a wavelength of 16–20 (so a frequency of 1/16–1/20, namely 1/8–1/10 of the distance from the center to the edge in the frequency domain). This correspondence demonstrates that our short-term memory model does manage to detect the dominant frequency pattern of the input image. The performance of the other integer threshold (1,2,3) is not shown in Figures 3.3 and 3.4 because we would like to have a contrasting color scale for higher thresholds. As expected, the performance decreases sharply from $f_0 = 4$ to $f_0 = 0$ which is demonstrated in Figure 3.5. Here, the threshold is increased by 1 unit.

From Figures 3.3 and 3.4, we note that our model is mostly noise-tolerant up to the noise level equal to 0.1. This manifests that our network is able to precisely recall very analogous patterns but has some trouble recalling the exact patterns for less analogous inputs. Another notable feature is that there exists a shift in performance pattern under different filter thresholds as the astrocytic coverage size is increased from $l = 2$ to $l = 9$, which stands as the central observation of our research, and next, we use $l = 4$ and $l = 8$ to explain it at greater length.

From Figures 3.3 and 3.4 we observe that for $l = 4$, the performance color transitions from light red to dark and back to light at a relatively high noise level. Take for example noise level equal to 0.2 (Figure 3.6), a low filter threshold, $f_0 = 10$ smooths the picture and prevents over-firing of neurons. A high filter threshold, $f_0 = 58$, ensures that most of the digital pixels are firing, although at the cost of slight over-firing. However, a middle one, $f_0 = 40$ corrupts the picture to a certain degree and yields a relatively low performance. For $l = 8$, the performance color transitions from light red to dark and there exists a slight recovery before going dark again at high noise levels. At the noise level equal to 0.2 (Figure 3.7), the firing patterns of $f_0 = 10$ and $f_0 = 40$ are very similar to those in $l = 4$, despite the alteration in the astrocytic coverage. However, for $f_0 = 58$, $l = 8$ significantly favors the over-firing which results in many misclassifications, and $f_0 = 50$ is somewhere in the middle.

In order to better summarize the results shown in Figure 3.3, we use box plots to exhibit our statistical analysis. Figure 3.8a displays the overall short-term memory performance subject to low salt-and-pepper noises by grouping the noise level from 0 to 0.1. Similarly, the performance subject to high salt-and-pepper noises is shown in Figure 3.8b by grouping the noise level from 0.12 to 0.2. We note that at the low noise level, the overall performance starts to decrease at $l = 6$ and there does not exist a significant change in performance when it comes to the high noise, although $l = 4$ and $l = 5$ have a higher median. Figure 3.9a displays the overall short-term memory performance subject to low filter thresholds by grouping the threshold from 4 to 22 and Figure 3.9b displays the performance subject to

high filter thresholds by grouping the threshold from 24 to 58. In both of them, we have witnessed a slight decrease in performance from $l = 6$. The cutting points of low and high in the above cases are chosen based on the patterns shown in Figures 3.3 and 3.4. However, if we investigate the performance subject to individual filter thresholds, it could look very different from what is shown in Figure 3.9. For instance, in Figure 3.10, at $f_0 = 46$ the best median is achieved at $l = 7$ whilst at $f_0 = 18$ there is a decrease in performance after $l = 6$ which is similar to the overall result (Figure 3.9a). This may raise the possibility that different sizes of astrocytic coverage might optimize the performance at different spatial frequencies. However, we need to point out that the number of data at individual thresholds is limited and therefore, the difference may not be as considerable as shown in the figures.

We also observe that for large astrocytic coverage, $l = 8$ for instance, a shorter time interval between training and testing tends to outperform a longer one when most of the frequency components have been included (Figure 3.11 right-end). Conversely, the performance barely changes with respect to a small astrocytic coverage such as $l = 4$ (Figure 3.12). For high filter thresholds, a longer time interval will result in more activated astrocytes as a result of calcium diffusion. However, a smaller astrocytic coverage has relatively little impact on the firing patterns of neurons at the testing stage because each astrocyte controls fewer neurons. Conversely, a larger coverage will result in the over-firing of neurons in that more neurons that should not be activated have been activated, which decreases the performance. The above analysis is supported by the calcium patterns of astrocytes with different astrocytic coverage sizes and time intervals between training and testing shown in Figure 3.13. For relatively low filter thresholds, the firing pattern remains largely unchanged because of smoothing, as demonstrated previously.

## 3.4 Discussions

In light of the results shown in Figures 3.3, 3.4, 3.9 and 3.10, there are two hypotheses that may sound plausible:

1. Astrocytes may adjust their coverage areas in response to the change in spatial

frequencies in order to optimize the short-term memory.

2. Different astrocytes may have different coverage areas in order to process different frequency components in order to optimize the short-term memory.

To the best of our knowledge, these open questions have not been given enough consideration yet and therefore, our work aims to raise the awareness of these plausible relations so that interested researchers may test and verify them in the laboratory. Hypothesis 1 and 2 are not identical but they are somehow similar and could co-exist. As shown in Figures 3.9 and 3.10, although the majority of $l = 4$ outperforms $l = 7$, at some specific thresholds ($f_0 = 46$ for instance) $l = 7$ gives a slightly better performance. This may raise an open question for experimentalists to validate whether astrocytes adjust their coverage areas in response to the changing spatial frequencies (hypothesis 1), or whether different astrocytes have different coverage areas (hypothesis 2), so as to assist the short-term memory. More precisely, it may be plausible to hypothesize that individual astrocytes are free to select from a wide range of coverage areas in order to optimally process the spatial information containing diverse frequency components; or at a particular time point, individual astrocytic modules, in which all astrocytes have identical coverage, process some particular frequency components and hierarchically summarize the information to achieve the optimal short-term memory. In all, one is interpreted from a dynamic viewpoint and the other one from a static viewpoint, but they do not contradict each other. The former one may also help to explain the findings that the astrocytic coverage of synapses is highly dynamic. Additionally, over the last decade, emerging evidence has shown that astrocytes actively participate in the brain energy mechanisms and potentially assist the energy-efficient coding of neuronal circuits [229–231]. It is reasonable to reckon that a compactly connected astrocytic network tends to consume more energy, therefore it seems plausible that a sparse layout could become more favorable so long as the precision is not considerably compromised. The results may provide a new perspective for those who study the roles played by astrocytes in the cerebral energy-efficiency.

In regards to the small noise level, on the whole, a small astrocytic coverage

tends to outperform a big one irrespective of the filter threshold. This may indirectly support the experimental result that the increasing density of astrocytes enhances short-term memory performances [199]. The comparison of $l = 2$, $l = 3$ and $l = 4$ also raises the potential to study whether over-crowded astrocytes will have negative effect on short-term memory for biologists.

As for the relatively high noise level, it appears that the performance score remains relatively low in a threshold interval and the more noisy the image is, the wider the interval is. We suppose that the phenomenon is due to the fact that a higher salt-and-pepper noise distorts the original image more massively and the image decomposes into more frequency components (including many high frequencies). A relatively high filter threshold retains these frequencies (as a result of noise) which leads to a decrease in performance.

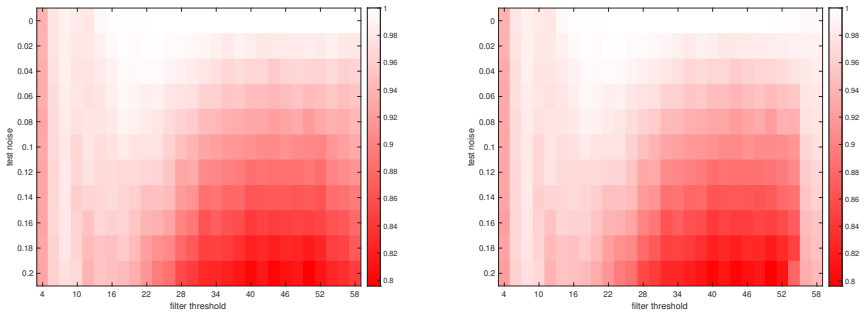
The astrocytic layer is bittersweet when it comes to short-term memory. On the one hand, the slow time scale (seconds) of calcium events in astrocytes enables the storage of the input pattern in the astrocytic layer and therefore, assists with the formation of short-term memory. On the other hand, the enhanced noise in the astrocytic layer may also corrupt the shape of the input pattern. As demonstrated in the previous section, the noise may be categorized into two classes. First, the neurons in the neuronal layer that fire incorrectly will translate the noise to the astrocytic layer and some astrocytes may be incorrectly activated. Second, the diffusion of the calcium event may also result in the activation of astroyctes that ought to remain silent. However, one should note that the results shown in Figure 3.11 were only based on a specific scenario of a toy example. It is still unknown how the diffusion between astrocytes will affect short-term memory in reality.

Finally, the sensitivity to the filter threshold also validates the necessity of introducing convolutional layers in spiking neural networks [232–234] for pattern recognition tasks because the idea of introducing filters is to extract the local patterns such as curves and straight lines.

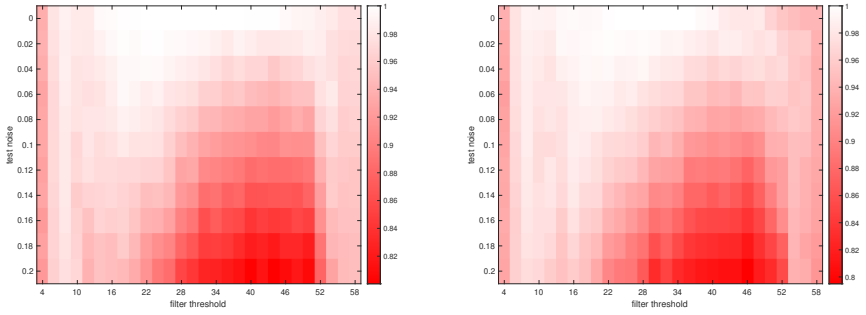
One open question is whether the pattern displayed in Figure 3.3 will scale up with the size of the input image. Namely, when the size of the input is scaled up

or down, whether the same pattern will be observed when the astrocytic coverage alters with the same ratio. This may shed light on the correspondence between the size of the input image and the astrocytic coverage.

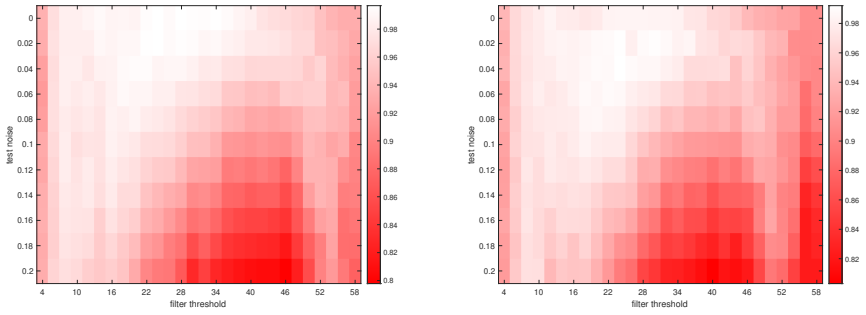
To summarize, in this work, we leveraged a computational neuron–astrocyte model for short-term memory that has been recently developed to study the impact of astrocytic coverage and spatial frequencies on short-term memory. We demonstrated a shift in the performance of the short-term memory under different filter thresholds as the astrocytic coverage size is altered. We also raised two hypotheses about the potential relationship between astrocytic coverage, spatial frequencies and short-term memory. These hypothetical results emerged from several experimental facts which show that the rearrangement of the structural interactions between synaptic elements and perisynaptic astrocytic processes alters the efficacy of neurotransmitter transport and gliotransmitter release, thereby inducing changes in the synaptic gain and long-term potentiation induction [235, 236]. However, the role of astrocytic morphological plasticity in memory processes required further experimental evaluation including cellular and in vivo studies. In particular, different genetic-interference strategies which impact the plasticity of structural interactions between astrocytes and synapses can be used to monitor memory impairments in animal studies [205]. We expect that the article can bring these unattended aspects to the attention of biologists as a better understanding of this topic may pave the way for some transformative findings as to how neurons and glial cells adapt their behaviors in response to external stimuli.



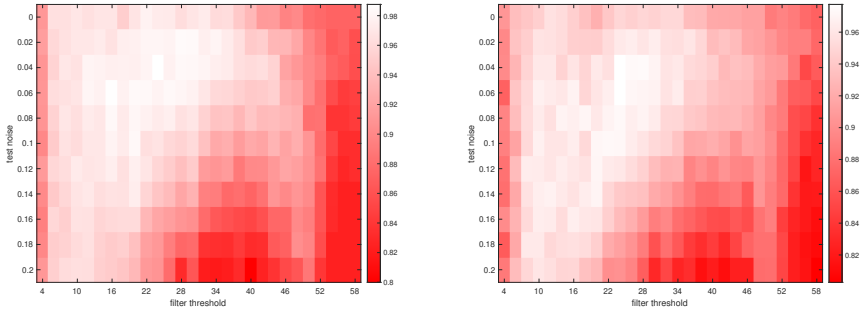

(a) $l = 2$ (b) $l = 3$




(c) $l = 4$ (d) $l = 5$

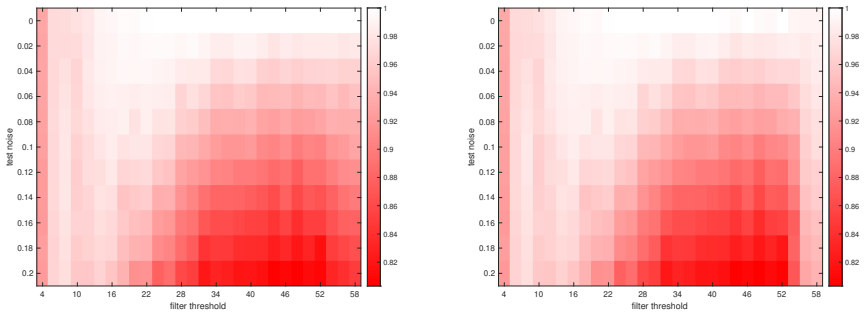



(e) $l = 6$ (f) $l = 7$




(g) $l = 8$ (h) $l = 9$

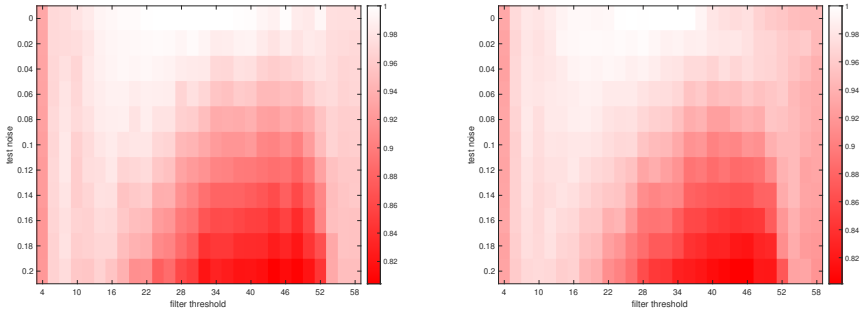
**Figure 3.3:** Performance score $C_p$ for digit Zero. In each sub-figure, the vertical axis denotes various levels of the salt-and-pepper noise at the testing stage. The horizontal axis denotes the threshold $f_0$ (increased by 2 units) of the low-pass filter. Each square represents the performance score $C_p$ obtained from the simulation described in Section 3.2.6, when a specific noise level is introduced in the testing stage and a specific low-pass filter threshold is applied at each time step. There exists a clear shifting pattern in performance as the size of the astrocytic coverage $l$ increases from 2 to 9, mainly in the lower right part of the figure.




(a) $l = 2$



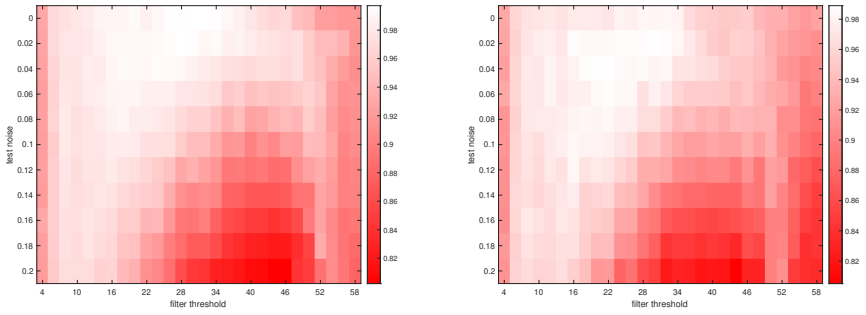

(b) $l = 3$




(c) $l = 4$



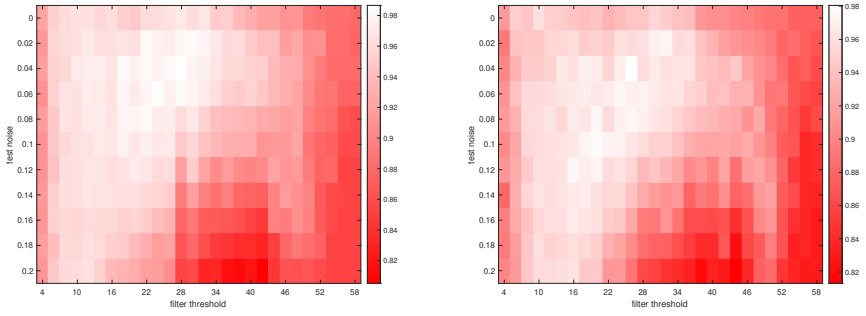

(d) $l = 5$




(e) $l = 6$




(f) $l = 7$




(g) $l = 8$




(h) $l = 9$

**Figure 3.4:** Performance score $C_p$ for digit Two. In each sub-figure, the vertical axis denotes various levels of the salt-and-pepper noise at the testing stage. The horizontal axis denotes the threshold $f_0$ (increased by 2 units) of the low-pass filter. Each square represents the performance score $C_p$ obtained from the simulation described in Section 3.2.6, when a specific noise level is introduced in the testing stage and a specific low-pass filter threshold is applied at each time step. There exists a clear shifting pattern in performance as the size of the astrocytic coverage $l$ increases from 2 to 9, mainly in the lower right part of the figure.

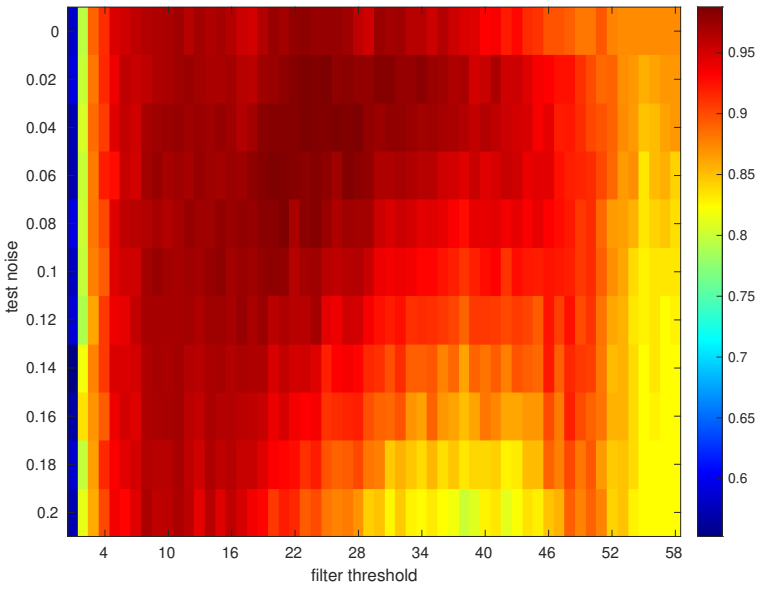
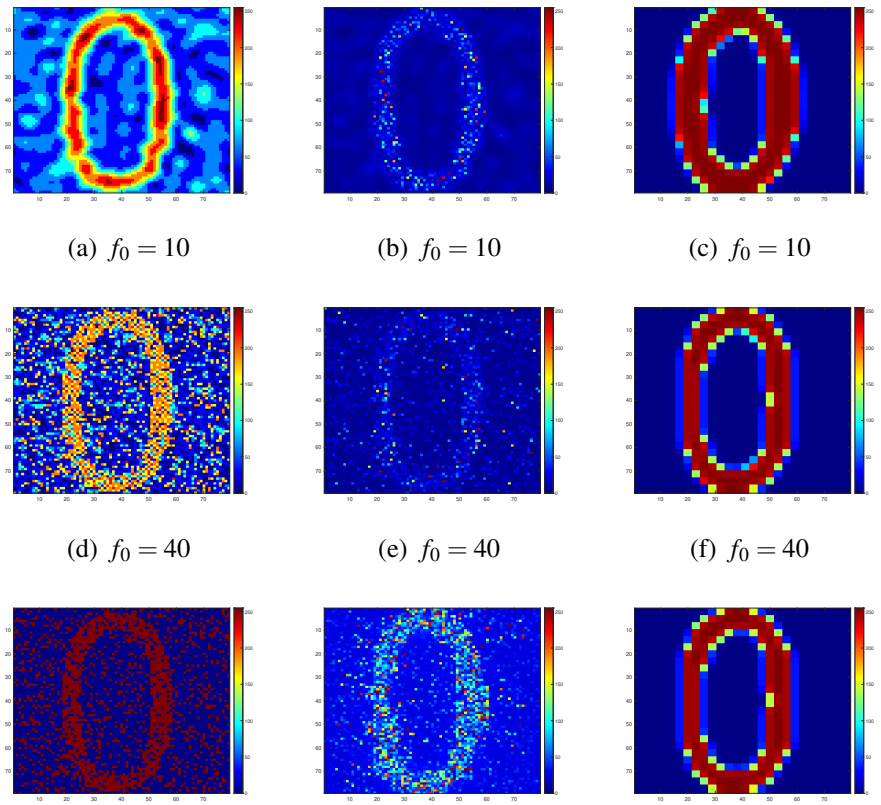



**Figure 3.5:** Performance score $C_p$ at diverse levels of noise with $l = 8$ for digit zero including all the whole-number low-pass filter thresholds. Each square represents the performance score $C_p$ obtained from the simulation described in Section 3.2.6, when a specific noise level is introduced in the testing stage and a specific low-pass filter threshold is applied at each time step. There is a sharp change in performance at threshold 4 and further explanation is elaborated in Section 3.3.

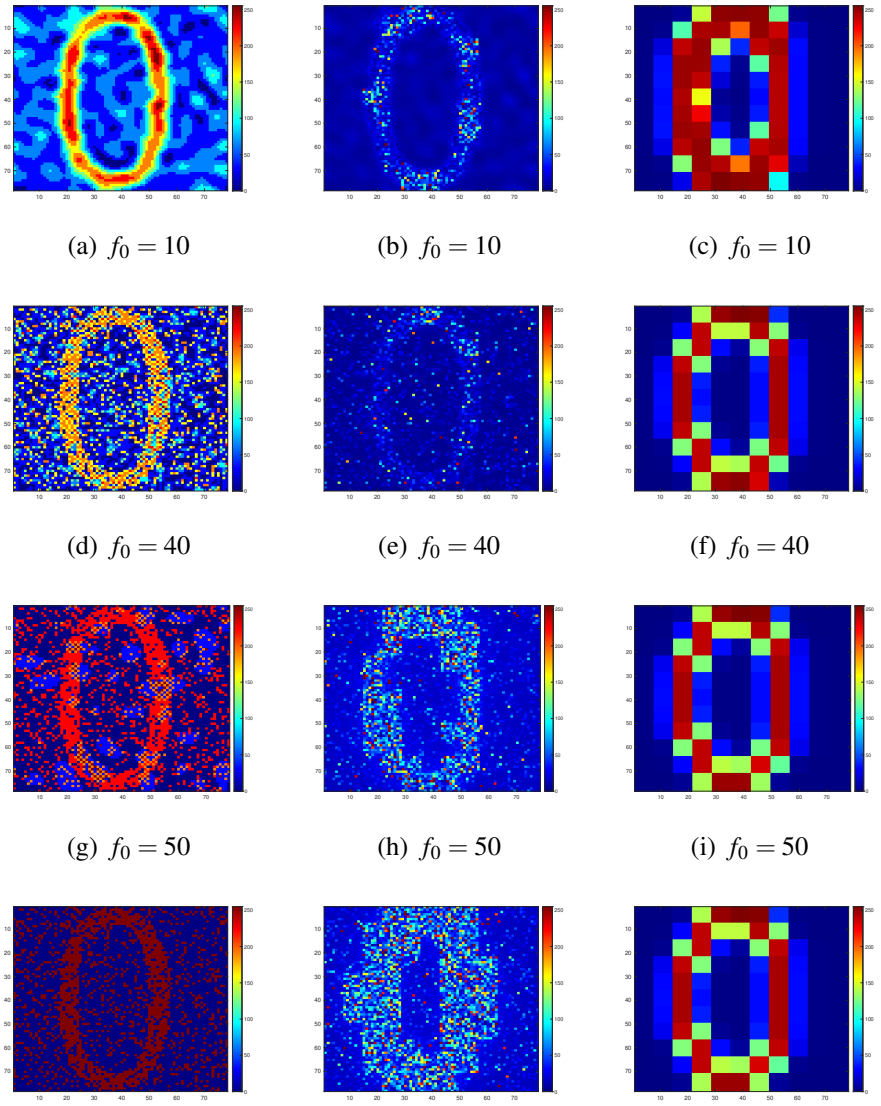


(a) $f_0 = 10$ (b) $f_0 = 10$ (c) $f_0 = 10$

(d) $f_0 = 40$ (e) $f_0 = 40$ (f) $f_0 = 40$

(g) $f_0 = 58$ (h) $f_0 = 58$ (i) $f_0 = 58$

**Figure 3.6:** Snapshots for $l = 4$. The **left** panel displays the input current after being transformed by the low-pass filter at testing. The **middle** panel displays the firing pattern of neurons at $t = 2.1$ s. The **right** panel displays the calcium pattern of astrocytes at $t = 2$ s. In each sub-figure, the x- and y-axis denote the positional indices of the image. The colorbar describes the level of the input current, neuronal firing and astrocytic calcium concentration. The level has been scaled in the range 0–255 for visualization. The model exhibits very similar calcium patterns irrespective of the filter thresholds. From the first and the second column, we observe that $f_0 = 10$ smooths the picture and prevents over-firing of neurons; $f_0 = 58$ ensures that most of the digital pixels are firing; $f_0 = 40$ corrupts the picture to a certain degree. Therefore, the performance of $f_0 = 40$ is less desirable.



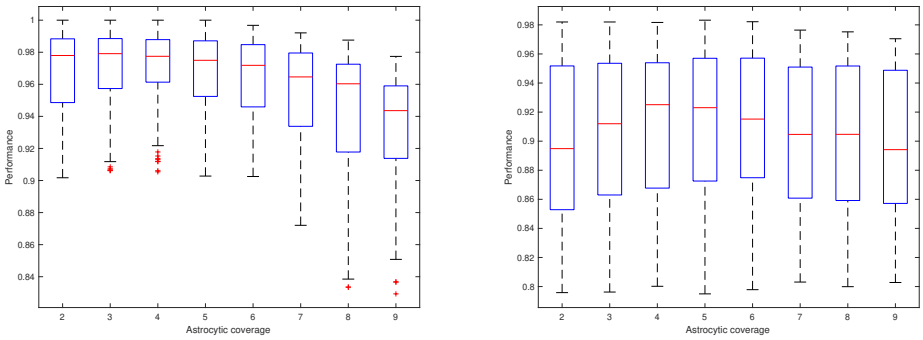
(a) $f_0 = 10$ (b) $f_0 = 10$ (c) $f_0 = 10$

(d) $f_0 = 40$ (e) $f_0 = 40$ (f) $f_0 = 40$

(g) $f_0 = 50$ (h) $f_0 = 50$ (i) $f_0 = 50$

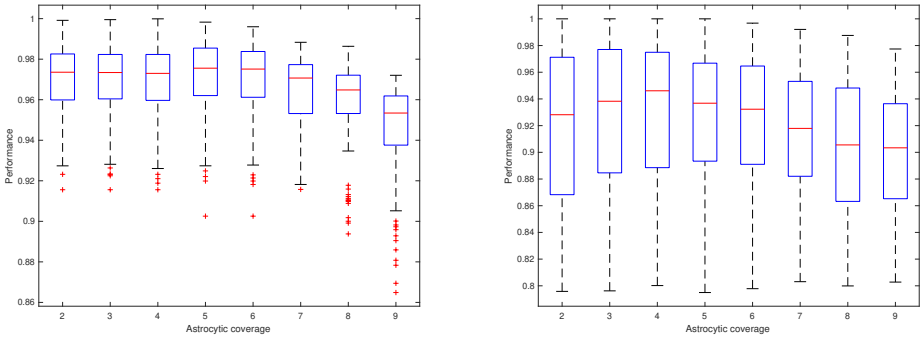
(j) $f_0 = 58$ (k) $f_0 = 58$ (l) $f_0 = 58$

**Figure 3.7:** Snapshots for $l = 8$. The **left** panel displays the input current after being transformed by the low-pass filter at testing. The **middle** panel displays the firing pattern of neurons at $t = 2.1$ s. The **right** panel displays the calcium pattern of astrocytes at $t = 2$ s. In each sub-figure, the x- and y-axis denote the positional indices of the image. The colorbar describes the level of the input current, neuronal firing and astrocytic calcium concentration. The level has been scaled in the range 0–255 for visualization. The model exhibits very similar calcium patterns irrespective of the filter thresholds. From the first and the second column, we observe that $f_0 = 10$ smooths the picture and prevents over-firing of neurons; $f_0 = 40$ corrupts the picture to a certain degree; $f_0 = 58$ results in over-firing; $f_0 = 50$ is somewhere in the middle. Therefore, the performances of $f_0 = 40$ and $f_0 = 58$ are less desirable.



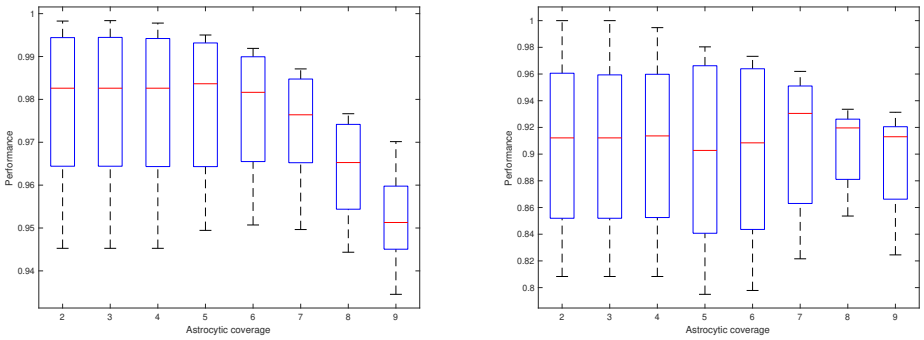

(a) Low noise (b) High noise

**Figure 3.8:** Short-term memory performance subject to low and high salt-and-pepper noises. Each box in the subfigure for the low noise is obtained by grouping the scores from 0 to 0.1 for all filter threshold values of the corresponding astrocytic coverage $l$ in Figure 3.3. The high noise is by grouping the scores from 0.12 to 0.2. The horizontal axis denotes the size of astrocytic coverage and the vertical axis denotes the performance score $C_p$. At the low noise level, the performance starts to decrease from $l = 6$; at the high noise level, there does not exist a significant change in performance, although the medians are slightly different.



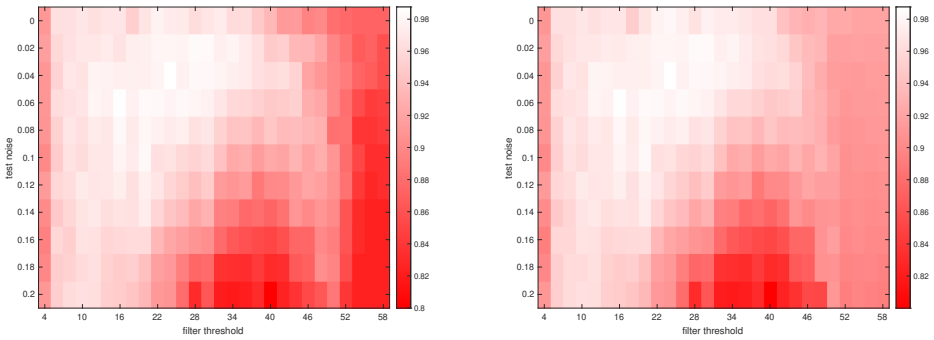

(a) Low-pass filter with low thresholds (b) Low-pass filter with high thresholds

**Figure 3.9:** Short-term memory performance subject to low-pass filter with low and high thresholds. Each box in the subfigure for the low thresholds is obtained by grouping the scores from 4 to 22 for all noise levels of the corresponding astrocytic coverage $l$ in Figure 3.3. The high noise is by grouping the scores from 24 to 58. The horizontal axis denotes the size of astrocytic coverage and the vertical axis denotes the performance score $C_p$. Both figures demonstrate a decline in performance from $l = 6$.



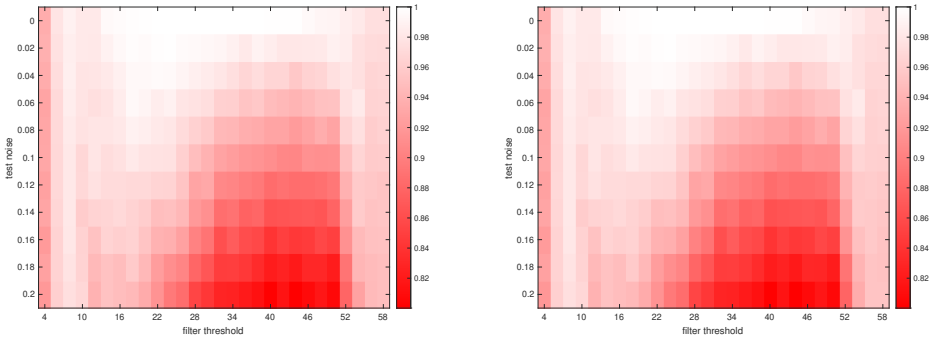

(a) $f_0 = 18$ (b) $f_0 = 46$

**Figure 3.10:** Short-term memory performance subject to two individual filter thresholds. Each box in the subfigure is obtained by grouping the scores for all noise levels of the corresponding astrocytic coverage $l$ in Figure 3.3. The horizontal axis denotes the size of astrocytic coverage and the vertical axis denotes the performance score $C_p$. At individual thresholds, $f_0 = 18$ is similar to the overall performance of the low thresholds; the highest median of $f_0 = 46$ is achieved at $l = 7$.



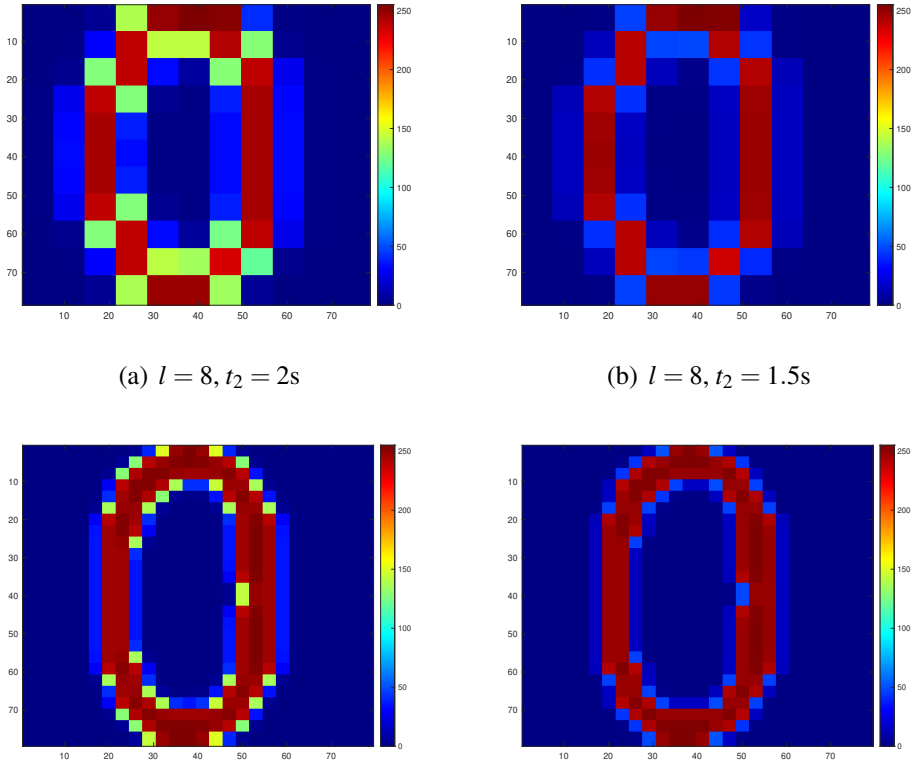

(a) $t_2 = 2\text{s}, t_e = 2.3\text{s}$ (b) $t_2 = 1.5\text{s}, t_e = 1.8\text{s}$

**Figure 3.11:** The performance score $C_p$ with different starting time $t_2$ with $l = 8$. In each sub-figure, the vertical axis denotes various levels of the salt-and-pepper noise at the testing stage. The horizontal axis denotes the threshold $f_0$ (increased by 2 units) of the low-pass filter. The right-end of (**a**) is darker than (**b**) and the remaining regions are nearly the same.




(a) $t_2 = 2\text{s}, t_e = 2.3\text{s}$ (b) $t_2 = 1.5\text{s}, t_e = 1.8\text{s}$

**Figure 3.12:** The performance score $C_p$ with different starting time $t_2$ with $l = 4$. In each sub-figure, the vertical axis denotes various levels of the salt-and-pepper noise at the testing stage. The horizontal axis denotes the threshold $f_0$ (increased by 2 units) of the low-pass filter. The performance is barely affected by the time interval.



(a) $l = 8, t_2 = 2\text{s}$ (b) $l = 8, t_2 = 1.5\text{s}$

(c) $l = 4, t_2 = 2\text{s}$ (d) $l = 4, t_2 = 1.5\text{s}$

**Figure 3.13:** Calcium patterns of astrocytes with different astrocytic coverage areas and time intervals between training and testing. Here, we use $f_0 = 58$ and noise level equal to 0.2 as an example.

# Chapter 4

# Novel efficient reservoir computing methodologies for regular and irregular time series classification

## 4.1 Introduction

Time series refers to a sequence of data points collected in chronological order over a period of time with each point typically being recorded at a specific timestamp. A time series has two main components, timestamp and observation. Timestamp presents the time at which a specific record is taken while an observation displays a value associated with each timestamp that informs the relative importance to the other time points. Additionally, time series data may come with some other patterns that makes the analysis of time series more challenging. For instance, samples from the same dataset may have different lengths (variable length) and/or adjacent time points may have different time intervals (heterogeneous interval). Time series analysis involves studying and interpreting patterns such as trends and dependency within the sample over time and has been widely applied to real-world phenomena [237–239]. Among them, time series classification (TSC) focuses on the task of categorizing and labeling sequential data into their distinct classes or categories and plays an indispensable role in medicine, telecommunications and finance. The efficacy of TSC algorithms relies on their capability of balancing short- and long-term

memory as well as capturing the time dependency, whilst distinguishing desired patterns from the noisy ones.

Over the past few decades, an astronomical amount of algorithms have been developed to address this particular field. Thus far, Long Short-Term Memory (LSTM) networks can be seen as a milestone breakthrough, offering a robust solution to the challenges posed by modelling complex long-term dependencies in sequential data [240–243]. LSTM networks are a type of recurrent neural networks (RNN) that take advantage of memory cells and gates as a means to control the flow of information through the network so that the vanishing gradient problems can be largely mitigated. However, the training of the network is enabled by the state-of-the-art backpropagtion through time (BPTT) techniques. While BPTT is a powerful and effective method, it can be computationally expensive, not least for large and deep neural networks. Apart from backpropagation-facilitated neural networks, distance-based methods have also demonstrated enormous success on a wide range of TSC tasks [244–246] and among them, 1-Nearest Neighbour Dynamic Time Warping (1NN-DTW) has been proven difficult to beat as compared to other similar methods [247]. Nevertheless, 1NN-DTW requires the computation of the pairwise distance between samples which can still substantially increase the computation overhead especially when the sample size is large. As a result, the quest for some more efficient methods has never ceased in an attempt to strike the right balance between accuracy and energy consumption.

Complex dynamical systems have demonstrated colossal potential in learning and computation in a wide spectrum of frameworks such as gene regulatory networks, cellular networks and artificial neural networks [248–252]. Among them, Reservoir Computing (RC) stands at the forefront of cutting-edge research in the field of machine learning and artificial intelligence, providing a promising approach to the challenges of processing complex temporal data [253–255]. It maps input signals to a non-linear high-dimensional dynamical system where neurons are recurrently connected, generating a comprehensive representation of the input features, and the training is only applied to the output layer which makes it a hyper-efficient

alternative to the mainstream deep neural networks including LSTM networks. Additionally, RC can also be regarded as a mini-brain, making it a more biologically plausible model and paving the way for better understanding the information processing in the brain [256–258]. Echo state networks (ESNs) are an instance of RC where sigmoid functions have been employed as the activation functions in the reservoir and have drawn growing attention by virtue of their ease of implementation and computational efficiency [259–261]. Nevertheless, most of these methods are focused on sequential predictions and the classification methods are still underdeveloped, partially down to the non-existence of BPTT training, in the sense that the loss at the terminal prediction may not be able to flow back and adapt the weights in the previous layer(s). Hence, in this work, we will propose two new paradigms for ESN that can efficiently perform TSC tasks whilst maintaining a desirable classification accuracy.

The chapter will be organized as follows: Section 4.2 will introduce two new paradigms that we proposed in the context of ESNs, in which Section 4.2.1 provides an overview of the ESNs, Section 4.2.2 presents a novel method that can address a wide range of datasets with regular time series whereas Section 4.2.3 presents an alternative to address the irregular ones. Section 4.3 presents the performance of our new methods and Section 4.4 includes the conclusion and discussion. Unfortunately, There does not exist a precise definition for *regular* and *irregular*. Loosely speaking, in this work, we refer to the sequences from the same dataset as *regular* if they have the same and a sizable length, and the intervals between the adjacent timestamps are even.

## 4.2 Methods

### 4.2.1 Echo state networks

Echo state networks (ESNs) are an instance of reservoir computing and the diagram is shown in Figure 4.1. An ESN generally consists of three layers, from the left to right are: an input layer, a reservoir layer and an output layer. The weights in the input and the reservoir layer are randomly created and fixed throughout the training

process and only the weights in the output layer are trainable. Here the weights in the input layer refer to the weights of the connections from the input to the reservoir layer.

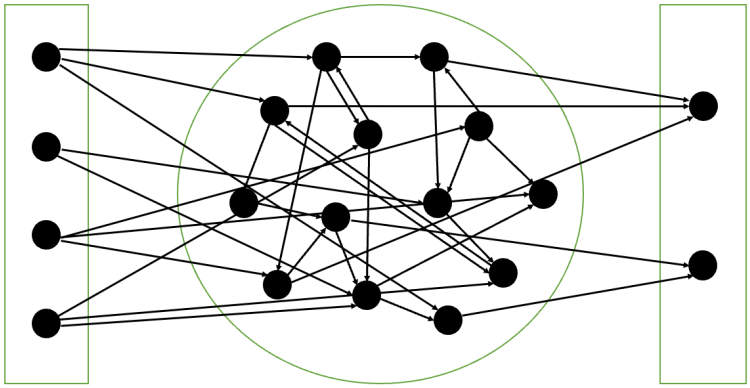


**Figure 4.1:** Echo state network (ESN)

First the input layer transforms the sequential data from the input layer into the reservoir layer using a randomly created input matrix. At the heart of an ESN is this fixed, large-scale recurrent reservoir of sparsely connected neurons. A distinct feature of ESNs is their echoic or fading memory property, mathematically known as the echo state property (ESP) which will be defined properly later. This characteristic allows ESNs to efficiently capture and retain relevant information from input sequences, making them particularly adept at handling time series data. In a standard setting, there are typically a lot more neurons in the reservoir layer than in the input layer in order for the reservoir to encode the input information into a high-dimensional dynamical system using sigmoid functions. The distribution used to generate the random connections will be introduced in due course. Then the temporal representation of the input signals within the reservoir layer can therefore be used to train the output weights given the teacher signals using a standard statistical model.

After introducing the basic notions of the ESNs, we will next define the general framework of an ESN using mathematical equations. Suppose a given longitudinal input signal has $k$ features and $T$ timestamps, the number of neurons inside the reservoir layer is $M$ and the number of neurons in the output layer is $C$. Here the

input signal at each timestamp is essentially a multidimensional data point. Then the sequential data at timestamp $t$ can be denoted by $\mathbf{u}_t \in R^k$, the internal state in the reservoir layer by $\mathbf{x}_t \in R^M$ and the output by $\mathbf{y}_t \in R^C$. Here $\mathbf{u}_t$ is simply the vector of the input values and $\mathbf{x}_t$ is the vector of values of all neurons within the reservoir layer. The internal state is updated by

$$\mathbf{x}_{t+1} = (1-a)\mathbf{x}_t + a\mathbf{tanh}(\mathbf{W}_{in}\mathbf{u}_{t+1} + \mathbf{W}_{res}\mathbf{x}_t + \mathbf{W}_{back}\mathbf{y}_t) \tag{4.1}$$

where $\mathbf{W}_{in}$ is the input weight matrix, $\mathbf{W}_{res}$ is a square matrix that represents the connections of the reservoir layer, $\mathbf{W}_{back}$ is the connections that project the output back to the reservoir and $a \in [0,1]$ is the leakage rate. **tanh** performs element-wise operation. However, as for the TSC tasks, the output is expected to be the predicted probabilities for distinct classes and normally only occurs at the terminal point which may not be informative of and compatible with the temporal dynamics in the reservoir layer. As a consequence, the feedback loop will be removed in our new paradigms and the equation will be reduced to

$$\mathbf{x}_{t+1} = (1-a)\mathbf{x}_t + a\mathbf{tanh}(\mathbf{W}_{in}\mathbf{u}_{t+1} + \mathbf{W}_{res}\mathbf{x}_t) = \mathbf{T}(\mathbf{x}_t, \mathbf{u}_{t+1}) \tag{4.2}$$

Apart from the size of the reservoir layer $M$, another pivotal factor is the leakage rate $a$ and it determines to what extent the internal state $\mathbf{x}_t$ is susceptible to the current input $\mathbf{u}_t$ and their neighbouring neurons. It has conventionally been chosen to be slightly less than 1. Lastly, the resulting output response at time $t$ can be described by

$$\mathbf{y}_t = f(\mathbf{W}_{out}[\mathbf{x}_t; \mathbf{u}_t]) \tag{4.3}$$

where $\mathbf{W}_{out}$ denotes the weight matrix in the output layer and $f$ is an arbitrary activation function. $\mathbf{W}_{out}$ is the only trainable part in ESNs and is normally trained by a linear classifier with input being the internal state $\mathbf{x}_t$ (sometimes together with $\mathbf{u}_t$) and output being the target of interest at timestamp $t$. Here we need to point out that in our new paradigms, the output layer will not be processed and trained as it stands which will be further clarified in due course.

The validity of an ESN is imparted by a basic stability property called the echo state property (ESP). It states that an ESN has the ESP if it can forget the initial values at a rate independent of the input, given any input sequence from a compact set [262]. A strict definition is provided below:

**Definition 1.** *An ESN has the echo state property if for any compact set A, there exists a null sequence* $(\delta_t)$, $t = 0,1,2,\ldots$ *such that for any input sequence* $\mathbf{u}_t \subset A$, $t = 0,1,2,\ldots$, $||\mathbf{x}_t^1 - \mathbf{x}_t^2|| \leq \delta_t$ *holds for any two initial states* $\mathbf{x}_0^1$ *and* $\mathbf{x}_0^2$ *and any* $t \geq 0$. *A null sequence is a sequence that converges to 0.*

To this end, $\mathbf{W}_{res}$ is scaled by a factor in order to largely ensure the ESP and details are included in Appendix C. Also for the sake of conciseness, we use the same notation $\mathbf{W}_{res}$ to denote the scaled matrix by default.

### 4.2.2 Method 1: Differential Echo State Networks

In this section, we will present our first paradigm for TSC tasks, Differential Echo State Networks (Diff-ESNs) and show that it can perform efficiently on a variety of datasets with regular time series.

The ESNs are based on the fact that the current internal state $\mathbf{x}_t$ is dependent on the last one $\mathbf{x}_{t-1}$ together with the current input $\mathbf{u}_t$. Let us assume that a dataset contains $N$ time series samples and the observation of sample $i$ at timestamp $j$ can be represented by $\mathbf{u}_{i,j} = [u_{i,j}^1, u_{i,j}^2, \ldots, u_{i,j}^k]^T$, where $i = 1,...,N$ and $j = 1,\ldots,n_i$. Here $n_i$ denotes the length of the sequence of sample $i$ and $k$ is the number of features in the dataset. Note that in our setting the first subscript denotes the sample index, the second subscript denotes the timestamp, and the superscript denotes the feature index. The outcome of patient $i$ is denoted by $y_i \in \{1,2,\ldots,C\}$ where $C$ is the total number of classes in this specific dataset.

One of the main obstacles that impedes the development of TSC techniques using ESNs is the invariable nature of the connections in most part of the ESN. Unlike sequential prediction tasks, where the output can be immediately linked to the input at each time step since the output is typically the prediction for the next observation $\mathbf{u}_{i,j+1}$ given $\mathbf{u}_{i,j}$, in TSC tasks, the class of a sample can only be inferred

after the time series reaches the terminal stage. Since the weights in the input and the reservoir layer are not adaptable in ESNs and as a result, the class information at the terminal point cannot effectively alter the dynamics in the reservoir layer on the basis of previous timestamps, an alternative route needs to be taken for the purpose of facilitating TSC while retaining the hyper-efficiency of ESNs. In this section, we will introduce a novel framework that leverages a bio-inspired neural coding method along with a hallmark of the dynamics in the reservoir layer to conduct TSC tasks.

The neural coding refers to the intricate way that the nervous system represents and processes information, specifically in the context of neurons and their activity patterns. There are several theories around neural coding mechanisms [263–265] and in spite of their controversy and validity, it seems plausible to boil the principles down to *Three S's*: Spikes, Sparsity, Static suppression, and a brief description is given as follows.

1. Spikes: It is widely believed that the cross-talk between biological neurons is enabled by spikes (action potentials or firings). Therefore, the input and output signals of a neuron may be reckoned to be a sequence of binary outcomes such as $0,0,1,0,0,0,0,0,1,0$.

2. Sparsity: Biological neurons spend most of the time in a silent state in order to minimize unnecessary energetic costs, which makes 0 far more likely to occur than 1.

3. Static suppression: It is also known as event-driven processing in plainer language. It has been demonstrated that the neuronal systems in the sensory periphery have the tendency to stimulate neuronal responses when subject to drastic spatial or temporal change in external stimuli [266, 267].

Since ESNs are artificial neural networks that employ the sigmoid activation functions to process information, it may not be appropriate to convert the continuous observations into spikes. However, we may still take advantage of principle 2 and 3 to make our new method more biologically plausible and heuristic. Instead of

using the original input sequence $\mathbf{u}_{i,1}, \mathbf{u}_{i,2}, \ldots, \mathbf{u}_{i,n_i}$ for each sample $i$, here the sequence will be transformed into their adjacent pairwise differences such that the new observation for sample $i$ at timestamp $j$ can be denoted by

$$\hat{\mathbf{u}}_{i,j} = \mathbf{u}_{i,j+1} - \mathbf{u}_{i,j}, \qquad j = 1,2,...,n_i - 1. \tag{4.4}$$

Also for the sake of conciseness, from now on, we will drop the hat on these new observations $\hat{\mathbf{u}}_{i,j}$ and the notations will stay the same in the context of ESNs. Now the information that the neurons in the reservoir layer have to learn has been transformed from the raw observations to the sharpness of change in magnitude of the adjacent observations. Besides, after the transform, the observations at many timestamps will lie in the vicinity of 0 which corresponds to the second principle listed above. In addition to its biological plausibility, abrupt changes in time series are reckoned to entail critical information of the fate of the sample and lie at the heart of time series analysis and as a result, using the differential values as in Equation 4.4 may be more informative than using the original sequence [268–270]. An example is given in Figure 4.2.

Now that we have properly defined the time series representation as the input to the ESN, we can now start formulating the model. After feeding the input sequence $\mathbf{u}_{i,1}, \mathbf{u}_{i,2}, \ldots, \mathbf{u}_{i,n_i-1}$ for participant $i$ into the ESN, the stimuli will update the internal state of the reservoir layer by

$$\mathbf{x}_{i,j+1} = (1-a)\mathbf{x}_{i,j} + a\mathbf{tanh}(\mathbf{W}_{in}\mathbf{u}_{i,j+1} + \mathbf{W}_{res}\mathbf{x}_{i,j}) \tag{4.5}$$

Note that the internal state should be reset to zero $\mathbf{x}_{i,0} = 0$ after feeding each sample so as to remove the unwelcome dependency between samples. Then we collect all internal states for each sample $i$, $\mathbf{x}_{i,1}, \mathbf{x}_{i,2}, \ldots, \mathbf{x}_{i,n_i-1}$ and next we need to seek a hallmark of the internal state that can be used as the features for the classifier. Here we use the variance of the sequence as the hallmark of the sample and the variance vector of sample $i$ can be denoted by $\mathbf{Var}(i) \in R^M$. Below is a brief reasoning for the choice of this particular hallmark. Sequences from different classes may have

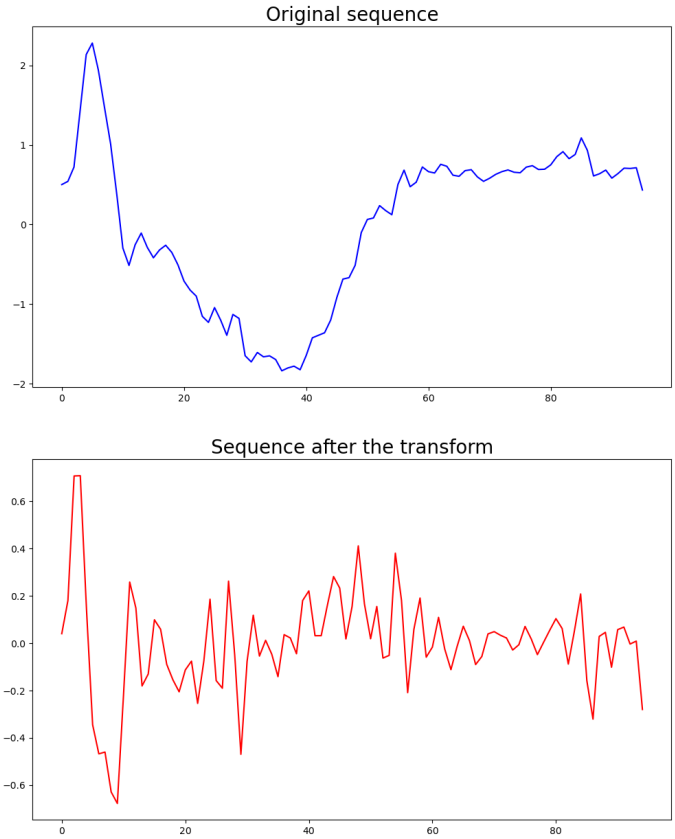



**Figure 4.2:** The sequence of a sample from the UCR ECG200 dataset. The upper panel exhibits the original sequence and the bottom exhibits the new sequence after being transformed by the differential operator. The x-axis indicates the timestamps and the y-axis indicates the magnitude of the observation at each timestamp.

distinct change levels at specific timestamps and the local level of variations can be captured by the variance to a certain extent. Thanks to the aforementioned coding method using the differential operator, the local variations can be more precisely reflected and fed into the reservoir layer. The reservoir layer generates rich and more complex dynamics capable of better retaining short-term memory such that the similar level of variation at distinct timestamps can be in a way, distinguished by separate neurons in the reservoir layer. Finally, the variances of all samples in

the training set will be stacked to form the input features to the classifier:

$$\mathbf{X}_{train} = \begin{bmatrix} \mathbf{Var}(1)^T \\ \vdots \\ \mathbf{Var}(i)^T \\ \vdots \\ \mathbf{Var}(N_{train})^T \end{bmatrix} = \begin{bmatrix} \mathbf{Var}^1(1) & \dots & \mathbf{Var}^M(1) \\ \vdots & \ddots & \vdots \\ \mathbf{Var}^1(i) & \dots & \mathbf{Var}^M(i) \\ \vdots & \ddots & \vdots \\ \mathbf{Var}^1(N_{train}) & \dots & \mathbf{Var}^M(N_{train}) \end{bmatrix} \tag{4.6}$$

Here $\mathbf{X}_{train}$ can be regarded as the design matrix in the regression model. Similarly, the corresponding classes will be stored in a vector $\mathbf{y}_{train}$:

$$\mathbf{y}_{train} = [y_1, \dots, y_i, \dots, y_{N_{train}}]^T \tag{4.7}$$

Then we fit $\mathbf{y}_{train}$ to $\mathbf{X}_{train}$ using the linear support vector machine (SVM) to train the classifier [271] as it is effective in high dimensional space, not least when the number of samples is not sufficient enough as compared to other statistical machine learning methods such as logistic regression. A quick treatment of SVM can be found in Appendix D. Lastly, the feature matrix and the classes of the test set can be defined in a similar way and we denote them by $\mathbf{X}_{test}$ and $\mathbf{y}_{test}$. In order to evaluate the performance of the model, we apply the trained classifier to $\mathbf{X}_{test}$, $\hat{\mathbf{y}}_{test} = CL(\mathbf{X}_{test})$ so that the outcomes of distinct samples can be predicted and compared with the ground truth outcomes. Here $\hat{\mathbf{y}}_{test}$ is the predicted class assignment and $CL$ is the classifier. The performance of the method will be shown in Section 4.3.1.

### 4.2.3 Method 2: Interpolation Echo State Networks

#### 4.2.3.1 Overview

As will be shown in Section 4.3.1, the Diff-ESNs have the potential to perform well when the time series data are regular. However, some datasets are more complicated than that due to the intrinsic difficulties in the data collection process. In some extreme cases, different samples (sequences) can have different numbers of observations with irregular time intervals whilst each sample may only have a hand-

ful of timestamps ($< 5$).

In this work, we use ovarian cancer screening to encapsulate the context of this particular data type and highlight the importance of tackling it. Ovarian cancer is a hereditary and lethal disease that disproportionately hits women aged above 50 and causes more than 150,000 overall deaths in the UK between 2017 and 2019, accounting for more mortalities than any other cancer arising from the female reproductive system. It is reported that the chance of a woman getting ovarian cancer is 1.3% and dying from it is 0.9% during her liftime. Whereas the 5-year survival rate is only around 40% due to late diagnosis, of which the majority of cases are diagnosed at stage III and IV, up to 90% of patients at stage I can be cured with conventional therapies, indicating the importance of early detection and intervention [272]. In order to address this long-standing health concern for ovarian cancer, various programs have been deployed worldwide which aim to discover early signs of cancer before the symptom appears when the medical intervention is more likely to be effective [273, 274]. Thus far, one of the most popular tests for ovarian cancer is the screening for tumour biomarkers. The participants have their samples taken several times in a time span of years so that their risks are well tracked and monitored. Some of them, unfortunately, will be diagnosed with cancer in the process of screening and will be transferred to medical treatment. As one can well imagine, the engagement of the participants is a highly spontaneous and independent behaviour which essentially makes the time series data exceptionally irregular. Furthermore, it is often too late for the treatment to kick in when the cancer is confirmed and as a result, early detection and prevention is also part and parcel to the wider population as well as to the optimization of the public health resources. Consequently, we will develop another method that can appropriately handle this type of datasets in the context of ESNs.

Unlike the Diff-ESNs, here the idea is to enhance the continuity of the time series such that more observations can be generated and the timestamps become more informative in the backdrop of the problem. For instance, if the biomarkers of a participant are registered at the age of 54.3, 54.9, 55.2 and 56.1, it might be of

interest to construct a more continuous trajectory with observations being recorded by month given that month is a meaningful unit in clinical study. Later, we will also show that it can enable flexible early forecasting, which is strongly preferable in cancer.

As a first step, linear interpolation is applied to the sequence of participant $i$, $\mathbf{u}_{i,1}, \mathbf{u}_{i,2}, \ldots, \mathbf{u}_{i,n_i}$ to fit the data into a continuous curve. Then the new sequence is acquired by sampling the interpolated curve on a monthly basis since the first record of screening and again for consistency, we will adopt the same notation to denote the new sequence for each $i$. An example is given in Figure 4.3. The curve exhibits the alteration of the expression of the biomarker $CA125$ from a specific participant. The blue dots signify the original data points and the crosses signify the new data points sampled by month after interpolation.

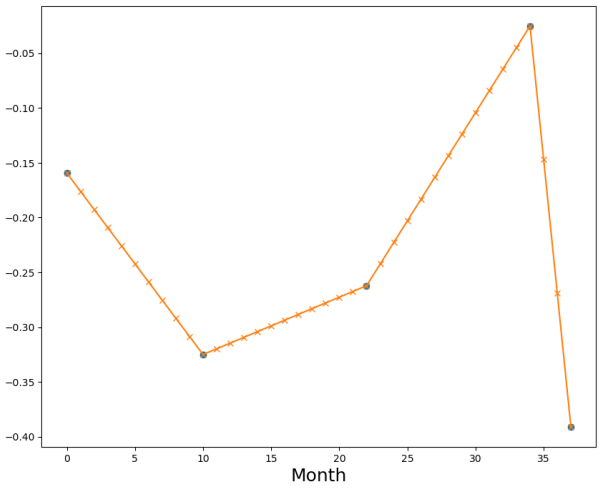



**Figure 4.3:** The linear interpolation of the biomarker $CA125$ of a specific participant. The x-axis stands for the month index since the start of the screening and the y-axis stands for the expression of the biomarker $CA125$. The blue dots are the original data points upon standardization and the crosses are the new data points sampled by month after interpolation.

Similar to the procedure in Section 4.2.2, we first collect all internal states for each sample $i$, $\mathbf{x}_{i,1}, \mathbf{x}_{i,2}, \ldots, \mathbf{x}_{i,n_i}$. In order to prepare the internal states as the input for a linear classifier, one needs to find a way to appropriately summarize the past history, not solely the last timestamp for each participant. There are effectively several options and here we will introduce two of them in the next two sections.

#### 4.2.3.2 Skip sampling approach

The first option is to stack all internal states in the training set according to the time series for individual participants into a matrix $\mathbf{X}_{train}$:

$$\mathbf{X}_{train} = \begin{bmatrix} x^1_{1,1} & \cdots & x^M_{1,1} \\ \vdots & \ddots & \vdots \\ x^1_{1,n_1} & \cdots & x^M_{1,n_1} \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ x^1_{i,1} & \cdots & x^M_{i,1} \\ \vdots & \ddots & \vdots \\ x^1_{i,n_i} & \cdots & x^M_{i,n_i} \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ x^1_{N_{train},1} & \cdots & x^M_{N_{train},1} \\ \vdots & \ddots & \vdots \\ x^1_{N_{train},n_{N_{train}}} & \cdots & x^M_{N_{train},n_{N_{train}}} \end{bmatrix} \tag{4.8}$$

And the corresponding outcomes will be stored in a vector $y_{train}$:

$$\mathbf{y}_{train} = [y_1, \ldots, y_1, \ldots\ldots, y_i, \ldots, y_i, \ldots\ldots, y_{N_{train}}, \ldots, y_{N_{train}}]^T \tag{4.9}$$

In a nutshell, provided that participant $i$ has $n_i$ timestamps, then there will be $n_i$ entries in $\mathbf{y}_{train}$ and $n_i$ rows in $\mathbf{X}_{train}$. However, we may not want to use all of them as training samples for two reasons:

1. The time span of a screening program is typically several years and after interpolating and sampling by month, most participants have dozens of timestamps. This may result in a substantial increase in the sample size and increase in the training overheads.

2. The internal state at timestamp $j$ for participant $i$, $\mathbf{x}_{i,j}$ may already contain a certain piece of the past information of the internal state, not least those

near the current timestamp $j$ by virtue of the recurrent structure. Including all timestamps may increase the learning variance and fail to generalize the patterns in the datasets.

Hence, we introduce an additional parameter $\tau$, where $\tau = 1, 2, \ldots, n_{min}$, to enable the option for users to sample the internal state $\mathbf{x}_{i,j}$ with skips. Here $n_{min} = min\{n_i\}_{i=1,2,\ldots,N}$ is the minimum length of time series among all participants. Since the marker expressions at the last timestamp may be of the utmost clinical relevance, we will always retain the internal state at the last timestamp for each participant in $\mathbf{X}_{train}$ and select rows in $\mathbf{X}_{train}$ backward with the skip $\tau$ such that the block for participant $i$ will become

$$\mathbf{X}_i = \begin{bmatrix} \vdots & \ddots & \vdots \\ x^1_{i,n_i-2\tau} & \cdots & x^M_{i,n_i-2\tau} \\ x^1_{i,n_i-\tau} & \cdots & x^M_{i,n_i-\tau} \\ x^1_{i,n_i} & \cdots & x^M_{i,n_i} \end{bmatrix} \tag{4.10}$$

The resulting design matrix therefore becomes

$$\mathbf{X}_{train} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_i \\ \vdots \\ \mathbf{X}_{N_{train}} \end{bmatrix} \tag{4.11}$$

And the corresponding subset of $\mathbf{y}_{train}$ will be taken to form the new outcome vector and again we keep the notation unchanged, $\mathbf{y}_{train}$. As such, enough samples can be ensured to train the classifier while taking into account the temporal dynamics. By training the internal states that incorporate diverse lengths of history (not just the last one) of a particular participant, it also potentially enables a better generalization when seeing the longitudinal features from other participants. Then we fit $\mathbf{y}_{train}$ to $\mathbf{X}_{train}$ using the linear SVM to train the classifier and generate the predicted outcomes for the test set $\hat{\mathbf{y}}_{test} = CL(\mathbf{X}_{test})$. Note that as opposed to the one shown

in Section 4.2.2, here the predicted outcomes $\hat{\mathbf{y}}_{test}$ are given in probability and the reason will be discussed at greater length later. As a result, the SVM with the probability output is implemented as a replacement [275]. Besides, at the moment, each participant still has multiple predicted outcomes and they are disparate for the same participant:

$$\hat{\mathbf{y}}_{test} = [\hat{y}_1^1, \ldots, \hat{y}_1^{n_1}, \ldots\ldots, \hat{y}_i^1, \ldots, \hat{y}_i^{n_i}, \ldots\ldots, \hat{y}_{N_{test}}^1, \ldots, \hat{y}_{N_{test}}^{n_{N_{test}}}]^T \tag{4.12}$$

One may need a single predicted outcome for each participant in order to be compared with the ground truth label. Here we select the prediction at the last time point to be the representative of the specific participant. In this way, we put more emphasis on the last time point as it not only contains arguably the most relevant information (the last screening), but also the history before that. In order to distinguish the approach from the one that we will show next, we name it *skip sampling*.

Early forecasting is instrumental to successful cancer treatment as it is often too late for the treatment to take effect when the cancer is confirmed. Therefore, lastly, we will demonstrate that the skip sampling approach can also enable flexible forecasting. The training process will remain the same and $\mathbf{X}_{train}$ will be employed as the input to the classifier. Suppose that we want to make the forecast $\gamma$ months before the last record. Instead of gathering all available observations, only the observations up to the timestamp $n_i - \gamma$ will be used as the input to the classifier in the

test stage:

$$\mathbf{X}_{test} = \begin{bmatrix} x^1_{1,1} & \cdots & x^M_{1,1} \\ \vdots & \ddots & \vdots \\ x^1_{1,n_1-\gamma} & \cdots & x^M_{1,n_1-\gamma} \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ x^1_{i,1} & \cdots & x^M_{i,1} \\ \vdots & \ddots & \vdots \\ x^1_{i,n_i-\gamma} & \cdots & x^M_{i,n_i-\gamma} \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ x^1_{N_{test},1} & \cdots & x^M_{N_{test},1} \\ \vdots & \ddots & \vdots \\ x^1_{N_{test},n_{N_{test}-\gamma}} & \cdots & x^M_{N_{test},n_{N_{test}-\gamma}} \end{bmatrix} \tag{4.13}$$

The remaining procedures stay the same and one can implement the skip sampling if needed.

#### 4.2.3.3 Average classifier approach

Thus far, we have presented a relatively generic approach since it does not demand any specific information about the nature of the dataset. As for the cancer data, it may be beneficial to take advantage of clinical observations to facilitate the algorithmic detection of cancer. The inception of tumor cell growth is widely recognized to occur well in advance of the manifestation of any noticeable. Therefore, a feasible idea is to train several classifiers with respect to different timestamps of interest and take the average of the respective classifiers. In mathematical terms, we first define

the matrices to account for each timestamp:

$$\mathbf{X}_{train}^{(j)} = \begin{bmatrix} x_{1,j}^{1} & \dots & x_{1,j}^{M} \\ \vdots & \ddots & \vdots \\ x_{i,j}^{1} & \dots & x_{i,j}^{M} \\ \vdots & \ddots & \vdots \\ x_{N_{train},j}^{1} & \dots & x_{N_{train},j}^{M} \end{bmatrix} \tag{4.14}$$

in which $\mathbf{X}_{train}^{(j)}$ collects the biomarker expressions at timestamp $j$ for all participants. In this work, we use $j = -1$ and $j = -6$ to collect input features for the separate classifiers because 6 months can be seen as a golden window in cancer diagnosis and treatment. We trained the two classifiers separately using the training set and we denote them by $CL_{-1}$ and $CL_{-6}$. Then the trained classifiers will be applied to their respective test sets $\mathbf{X}_{test}^{(j)}$, $j = -1, -6$, and the respective predicted outcomes will be generated such that $\hat{\mathbf{y}}_{test}^{-1} = CL_{-1}(\mathbf{X}_{test}^{(-1)})$ and $\hat{\mathbf{y}}_{test}^{-6} = CL_{-6}(\mathbf{X}_{test}^{(-6)})$. The final predicted outcomes (in probability) will be the average of the two

$$\hat{\mathbf{y}}_{test} = \frac{\hat{\mathbf{y}}_{test}^{(-1)} + \hat{\mathbf{y}}_{test}^{(-6)}}{2} \tag{4.15}$$

Again to make the distinction, we call this method *average classifier*.

## 4.3 Performance

### 4.3.1 Performance of the Differential Echo State Networks

The performance of the method can be assessed by computing the error rate in the test set of each dataset. Namely,

$$ER = \frac{\sum_{i=1}^{N_{test}} I(y_i \neq \hat{y}_i)}{N_{test}} \tag{4.16}$$

Here $N_{test}$ is the number of samples in the test set, $I$ is an indicator function, $y_i$ is the ground truth class of sample $i$ and $\hat{y}_i$ is the class category predicted by using the aforementioned trained classifier.

The UCR repository [276] is widely used as the benchmark datasets for assessing the performance of TSC algorithms. Each of these UCR datasets comprises a training set and a test set and the idea is to train any new model on the training set and report the error rate of the test set. In each dataset of this repository, all samples (sequences) have the same length and the lengths are all no less than 30 but can go up to several thousands for some datasets. Furthermore, all samples are recorded at the same timestamps in each dataset. In order to demonstrate the extensive application of our method, as a first step, the error rates of 33 datasets have been calculated and compared with the gold standard 1NN-DTW method. The parameters used for this study are $M = 50$ and $a = 0.9$. The connections in the input and the reservoir layer are fixed random matrices following the standard normal distribution. We will show that employing a small reservoir layer ($M = 50$ and hence with superb efficiency) and a commonly used leakage rate $a$ can lead to adequate classification accuracy. The robustness of the method will be evaluated later.

Table 4.1 displays the error rates generated by the Diff-ESN method, with a reference to the publicly available results using 1NN-DTW. The columns from the left to the right are the name of the dataset, number of classes, training size, test size, the error rate of using the 1NN-DTW and the Diff-ESN. The error rates in the last column will be highlighted in bold colour if they outperform the 1NN-DTW method. Some dataset names are abbreviated to fit the window for better visualization. As can be seen, the Diff-ESN achieves comparable results on all these 33 datasets relative to the 1NN-DTW. Among them, the Diff-ESN outperforms the 1NN-DTW on 23 datasets and 19 of them are below 0.25. Furthermore, all these tasks can be completed on a personal computer with minimal costs ($< 5$ minutes for the most training expensive dataset without even exploiting the parallel processing).

Since the connections in the input and the reservoir layer are randomly created, it is also imperative to validate the robustness of our new method. To this end, we select four datasets and have a closer look at the variation in performance subject to different connections. Figure 4.4 lays out the error rates of the dataset ECG200, Plane, ProxPhalOutAgeGrp and ShapeletSim produced by 50 different connections.

As illustrated, the performance is barely susceptible to the change in connections. Take the dataset ECG200 for example, as a matter of fact, the reported result in Table 4.1 (0.24) is located in the upper side of the violin plot in the sense that the majority of the connections will give rise to a lower error rate than 0.24.

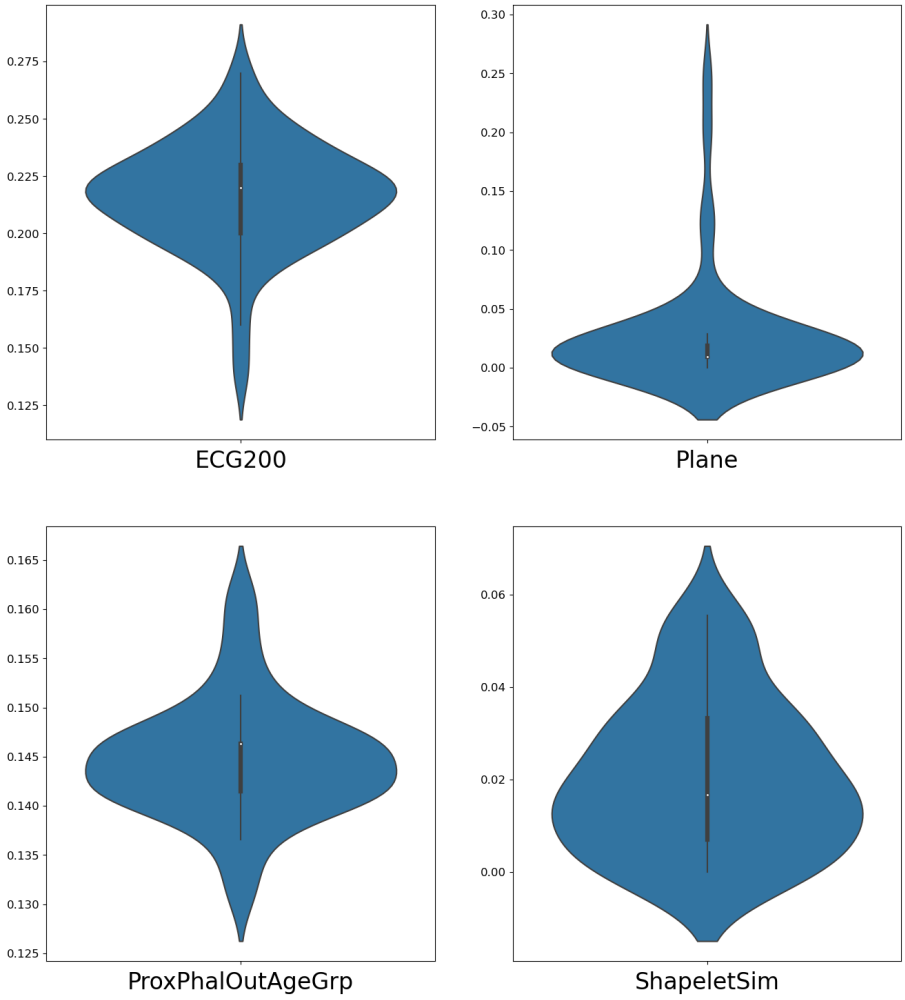



**Figure 4.4:** The robustness check for the Diff-ESN using the dataset ECG200, Plane, ProxPhalOutAgeGrp and ShapeletSim. Each violin plot contains 50 error rates generated by different random connections in the input and the reservoir layer. The vertical axis denotes the error rate.

### 4.3.2 Performance of the Interpolation Echo State Networks

The error rate (or accuracy in reverse) is an intuitive measure of correctness of statistical models. The UCR benchmark datasets have been extensively studied by the wider community and the training and the test sets have been carefully pre-split for a more perspective comparison and therefore, in Section 4.3.1, we compute the er-

**Table 4.1:** Error rates of UCR datasets using the Differential ESN. The columns from the left to the right are the name of the dataset, number of classes, training size, test size, the error rate of using 1NN-DTW and Diff-ESN. The error rates in the last column will be highlighted in bold colour if they outperform the 1NN-DTW method. Some dataset names are abbreviated to fit the window. The unavailable results will be denoted by /.

| Dataset | #class | #train | #test | 1NN-DTW | Diff-ESN |
|---|---|---|---|---|---|
| BeetleFly | 2 | 20 | 20 | 0.300 | **0.250** |
| BirdChicken | 2 | 20 | 20 | 0.250 | **0.100** |
| CinCECGTorso | 4 | 40 | 1380 | 0.349 | **0.336** |
| Coffee | 2 | 28 | 28 | 0 | 0.070 |
| DistPhalanxTW | 6 | 400 | 139 | 0.290 | 0.324 |
| Earthquakes | 2 | 322 | 139 | 0.258 | **0.252** |
| ECG200 | 2 | 100 | 100 | 0.230 | 0.240 |
| ECG5000 | 5 | 500 | 4500 | 0.250 | **0.081** |
| ECGFiveDays | 2 | 23 | 861 | 0.232 | 0.245 |
| ElectricDevices | 7 | 8926 | 7711 | 0.399 | 0.404 |
| FordA | 2 | 3601 | 1302 | 0.438 | **0.120** |
| FordB | 2 | 3636 | 810 | 0.406 | **0.328** |
| Ham | 2 | 109 | 105 | 0.533 | **0.377** |
| Herring | 2 | 64 | 64 | 0.469 | **0.406** |
| ItalyPowerDemand | 2 | 67 | 1029 | 0.050 | 0.076 |
| MidPhalOutCorr | 2 | 600 | 291 | 0.352 | 0.354 |
| MidPhalanxTW | 6 | 399 | 154 | 0.416 | 0.442 |
| OSULeaf | 6 | 200 | 242 | 0.409 | **0.380** |
| Plane | 7 | 105 | 105 | / | **0.029** |
| ProxPhalOutAgeGrp | 3 | 400 | 205 | 0.195 | **0.141** |
| ProxPhalOutCorr | 2 | 600 | 291 | 0.216 | 0.258 |
| ProxPhalanxTW | 6 | 400 | 205 | 0.263 | **0.195** |
| RefrigeDevices | 3 | 375 | 375 | 0.536 | **0.528** |
| ShapeletSim | 2 | 20 | 180 | 0.350 | **0.039** |
| SmallKitchenAppl | 3 | 375 | 375 | 0.357 | **0.307** |
| SonyAIBORobotSurf1 | 2 | 20 | 601 | 0.275 | **0.170** |
| SonyAIBORobotSurf2 | 2 | 27 | 953 | 0.169 | **0.127** |
| StarLightCurves | 3 | 1000 | 8236 | 0.093 | **0.089** |
| ToeSegmentation1 | 2 | 40 | 228 | 0.228 | **0.180** |
| Trace | 4 | 100 | 100 | 0 | 0.020 |
| TwoLeadECG | 2 | 23 | 1139 | 0.096 | **0.001** |
| Wafer | 2 | 1000 | 6164 | 0.02 | **0.015** |
| Worms | 5 | 181 | 77 | 0.536 | **0.442** |

ror rate of the test set and compare it with the publicly available results. However, in many other cases, it is more advantageous to generate the receiver operating char-

acteristic curve (ROC curve) and calculate the area under the curve (ROC AUC in short form) in order to evaluate the discrimination power of the model. It comes with a few reasons. Firstly, as opposed to the error rate, the ROC curve examines all possible classification which accurately reflects the model's response to the alteration of the threshold value. Secondly, the ROC curve is less susceptible to the imbalanced datasets. It is often the case that the clinical data are imbalanced since negative outcomes are way more likely to occur than positive outcomes. Subsequently, the error rate may still be low even if the model under-performs on the minority class and this can be substantially alleviated by the ROC curve. A detailed illustration of the ROC curve is provided in Appendix E.

In this study, the dataset that we used to assess the performance of our model is the BD dataset [277–279]. The BD dataset contains 222 patients on screening for ovarian cancer after removing those with only one timestamp. The biomarkers of interest are CA125, Glycodelin, HE4, MSLN25, MMP745 and CYFRA55, the expressions of which have been standardized before fitting any machine learning models. Among them, CA125 is a protein that has been regarded as the primary marker for ovarian cancer and the elevated levels of CA125 can be associated with certain conditions [280, 281]. The recorded times of these marker expressions are also included in the dataset which enables the interpolation of the time series.

With a view to evaluating the performance of our method, the BD dataset is split 50-50 randomly in that half of the dataset is used for training and the other half for testing. Considering that the BD dataset is a new and relatively small dataset, the robustness of the model in regard to the dataset needs to be checked carefully. With this goal in mind, 50 different splits of the dataset will be studied so as to gain a more comprehensive view of the model. Additionally, multiple values of the number of the neurons in the reservoir layer, $M$, will also be explored since changing the value of $M$ will not only examine this important parameter, but also alter the connections in the input and the reservoir layer so that more random connections can be inspected at the same time.

Figure 4.5 displays the ROC AUC scores using the skip sampling method with

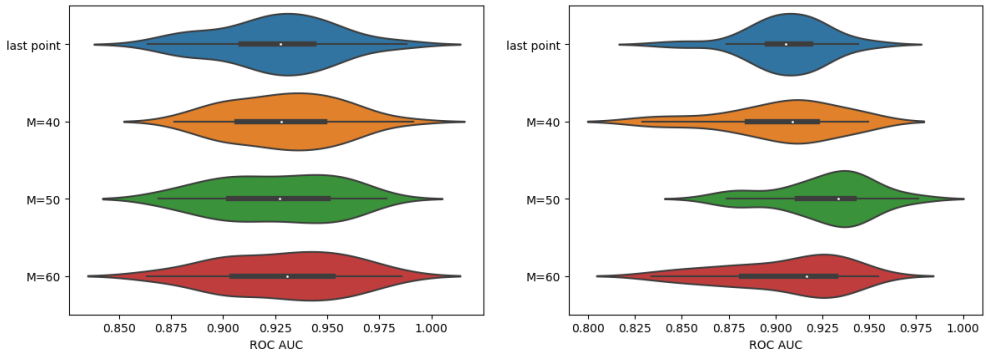
$\tau = 4$ with (a) being the case where only CA125 is used as the input and (b) where all 6 markers are included. Each figure contains 4 violin plots and each violin plot exhibits 50 AUC scores that correspond to 50 different splits of the training and the test set. In order to better evaluate the discrimination power of our time series based method, a comparison has also been made with the scores using only the last time point. In the latter case, only the last recorded marker expressions are selected for each participant and the logistic regression model is fitted to the training set and the outcomes (in probability) of the participants are predicted in the test set accordingly. The first row of each figure shows the AUC scores obtained by using the aforementioned logistic regression model as a baseline result. Row 2-4 show the AUC scores obtained by the skip sampling method with $M = 40, 50, 60$, respectively. As can be seen, most splits give rise to a consistently high ROC AUC score ($> 0.9$), irrespective of the markers involved. In the case where only CA125 has been used, $M = 60$ yields a slightly better overall performance than the baseline while $M = 40$ and $M = 50$ also reach a comparable level (and arguably better in some aspects). In the case of the 6 markers, $M = 40$, $M = 50$ and $M = 60$ all give rise to a higher overall performance and among them, the improvement of $M = 50$ on the baseline result is highly significant.

As for the average classifier, another 13 participants with only two registered timestamps will also be removed to ensure that the 6-month record from the last exists for each participant. Figure 4.6 displays the ROC AUC scores using the average classifier method. Similar to the skip sampling method, most splits give rise to a consistently high ROC AUC score. For CA125, $M = 40$ results in a notably better performance while $M = 50$ and $M = 60$ both reach comparable performance. For the 6 markers, $M = 50$ results in a marginally higher median while $M = 40$ achieves a comparable performance in terms of median and inter-quartile. Also note that the overall performance is marginally lower than the skip sampling method which could boil down to the fact that more participants are removed from the dataset and consequently, few samples are included in the training set.

Finally, the performance of the 6-month forecast is presented in Figure 4.7

using the skip sampling approach. As before, only the participants with at least three timestamps will be included to ensure the existence of the 6-month window. As shown, most splits give rise to a reasonably high ROC AUC score ($> 0.75$) and the majority of them lie above 0.8, which demonstrates a robust forecasting power of our method.

The reason that we want to assess the performance of different parameters ($M$) is that, as compared to using the cross validation and selecting the optimal parameter from the training set, it allows for a better inspection of the robustness subject to different parameters, which is reckoned crucially important for any new methodologies.



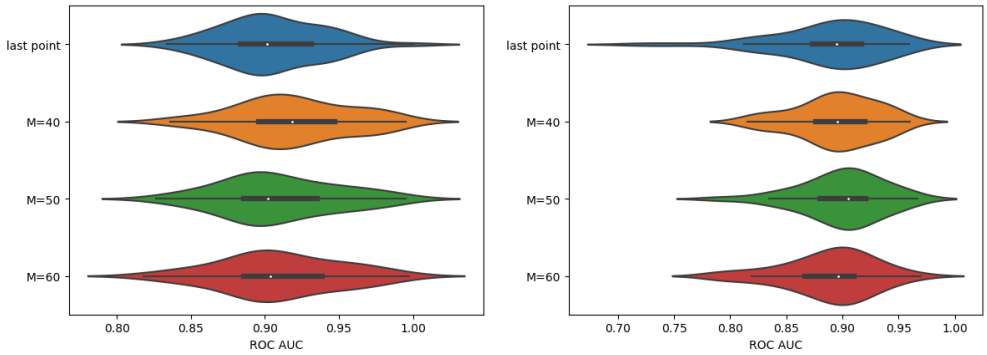

(a) CA125 (b) 6 markers

**Figure 4.5:** ROC AUCs of the skip sampling method. (a) shows the ROC AUC scores using only the marker CA125 and (b) shows the scores using the marker CA125, Glycodelin, HE4, MSLN25, MMP745 and CYFRA55. Each violin plot contains 50 AUC scores obtained from 50 different splits of the dataset. Each figure contains four violin plots. The first row shows the reference scores using the logistic regression model only on the last timestamp. Row 2-4 show the scores with different $M$ respectively. The x-axis denotes the ROC AUC score.

## 4.4 Conclusion and discussion

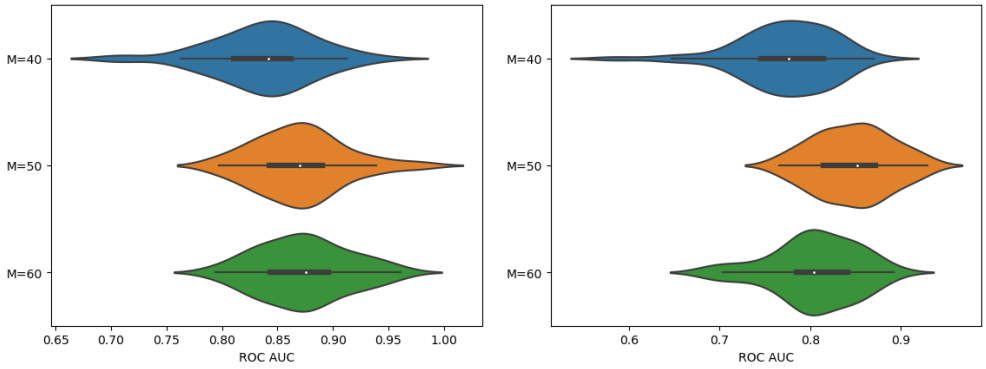
In this work, we established a novel ESN framework that can effectively address TSC tasks, comprising standard benchmark datasets as well as the more challenging cancer screening dataset, with tremendous efficiency.

We demonstrated that the Diff-ESN method has attained comparable performance with the classic 1NN-DTW method and resulted in notable improvement




(a) CA125 (b) 6 markers

**Figure 4.6:** ROC AUCs of the average classifier method. (a) shows the ROC AUC scores using only the marker CA125 and (b) shows the scores using the marker CA125, Glycodelin, HE4, MSLN25, MMP745 and CYFRA55. Each violin plot contains 50 AUC scores obtained from 50 different splits of the dataset. Each figure contains four violin plots. The first row shows the reference scores using the logistic regression model only on the last timestamp. Row 2-4 show the scores with different $M$ respectively. The x-axis denotes the ROC AUC score.




(a) CA125 (b) 6 markers

**Figure 4.7:** ROC AUCs of the 6-month forecast using the skip sampling method. (a) shows the ROC AUC scores using only the marker CA125 and (b) shows the scores using the marker CA125, Glycodelin, HE4, MSLN25, MMP745 and CYFRA55. Each violin plot contains 50 AUC scores obtained from 50 different splits of the dataset. Each figure contains three violin plots and they show the scores with different $M$ respectively. The x-axis denotes the ROC AUC score.

on error rate on several datasets, as shown in Table 4.1. The Interp-ESN method has been designed to tackle the famously hard irregular time series such as cancer screening data where forecasting on a regular basis is desirable. All cases that we considered as in Figure 4.5 and 4.6 achieve at least no worse overall performance

than the baseline using the logistic regression on the last recorded time point. Some parameters have given rise to a significant increase in performance in terms of aspects such as median and inter-quartile. Most importantly, the method entails a flexible forecasting option that empowers the prediction of the outcome at any month in advance and attains high ROC AUC scores with a 6-month forecast.

With the contribution being said, our model comes with several limitations and some further considerations can be given in the future. First of all, the Diff-ESN may not be an appropriate approach for the irregular time series such as the cancer screening data. It is because the interpolated sequence only yields a predicted coarse trajectory and the differential operator will compute inaccurate difference between two adjacent timestamps. Therefore, the hallmark of the trajectory that we chose, variance, may not be as predictive. Secondly, even though the Interp-ESN, not least the skip sampling approach, may be deemed as a more generic method, it comes with inevitable inconvenience when being applied to the regular time series data such as those listed in Table 4.1. The most prominent one is the pre-specified sampling step and its trade-off with computational efficiency. Even a sizable step size can still incur a relatively high computational cost when the time series sequence is exceptionally long and there is no guarantee that the selected sampling step is informative in any possible way. Lastly, there still exists a significant gap between the performance of our method reported in Table 4.1 and the state-of-the-art LSTM neural networks shown in [282] for most of the datasets. Yet, the deep neural networks require the training of an astronomical amount of parameters through back-propagation whereas our method is free of back-propagation and the training can be completed with a negligible energy cost. The future direction may lie in a more sensible trade-off between classification accuracy and energy consumption. One might consider introducing a light training algorithm for the weights in the reservoir layer as well as the output layer at each time step. This is also in compliance with the observation in Figure 4.4, 4.5 and 4.7 that some random connections lead to better performance than others. However, it is not very clear how to fix the computational complexity and whether the convergence will be guaranteed or not.

# Chapter 5

# When reservoir computing meets information theory: the tendency of entropy change through spike timing-dependent plasticity

## 5.1 Introduction

In Chapter 4, we have thoroughly discussed how ESNs, an instance of RC, can be potentially harnessed to accomplish TSC tasks and attain an accuracy comparable to some state-of-the-art methods with great efficiency. Another instance of RC, liquid state machines (LSMs), stand out as a promising and innovative approach in computational neuroscience, not least in understanding information processing and transmission [283–286]. LSMs leverage the dynamic properties of a liquid medium to enable robust and adaptive computations and can be regarded as a better-simplified brain by virtue of the utilization of biological neuronal models, which makes them a more biologically inspired framework, as opposed to the ESNs [92]. Because of this, it is also sometimes referred to as spiking RC. The fundamental mechanism behind LSMs is the use of a recurrent network, often referred to as the liquid consisting of interconnected biological neurons with random synaptic weights between them. This liquid medium exhibits rich temporal dynamics, allowing it to process

and transform incoming information in a highly parallel and distributed manner. The interactions among these neurons create a dynamic reservoir that serves as the foundation for the computation performed by the system.

Information theory is a discipline of applied mathematics that was first introduced in 1948 in an attempt to quantify information and to better unravel the myth of data transmission in telecommunication and biological systems [287–293]. It comprises numerous concepts and among them, entropy plays an indispensable role in uncovering the fundamental nature of information processing. In essence, entropy measures the uncertainty, or disorder inherent in data, offering profound insights into the efficiency of data compression, reliability of information transmission, and the intricacies of coding theory, etc. In particular, the complex relationship between the brain and entropy has attracted growing attention in recent years due to the increasing demand for deciphering the function of the brain [294–298]. A case in point is the hypothesis, the entropic brain, which seeks to establish the connection between entropy and the state of consciousness [299–301].

Aside from studying the information processing arising from different dynamics and spatial patterns at resting states, the response of the brain activity to external stimuli can also be investigated by means of ever-growing brain stimulation techniques. Amidst them, several works have already delved into the entropy change in the brain subject to various types of stimulation under miscellaneous physiological conditions and either increase or decrease in entropy was reported accordingly [89–91]. For example, [89] measured the Shannon entropy change in EEG and FEMG activity within a specific frequency range among patients during painful stimuli. [90] recorded the action potentials of neurons in the globus pallidus and the motor thalamus of monkeys rendered parkinsonian during high-frequency stimulation and clinically ineffective low-frequency stimulation, and the difference in Shannon entropy was computed accordingly. In [91], the sample entropy [302] of healthy patients was measured from resting state fMRI before and after applying high-frequency repetitive transcranial magnetic stimulation or control stimulation.

However, these experiments are confined to a specific setting and a global un-

derstanding of the trend of entropy change in the brain is still lacking and it is difficult to be carried out in a wet lab due to the cost and conditions. Therefore, mathematical models can be potentially useful for providing more insight into the phenomena of this kind and suggest plausible hypotheses from a global perspective provided that the model is appropriately constructed. In particular, we will attempt to answer a slightly different question to the experiments mentioned before. The emphasis will be on the tendency of entropy change when subject to stimulation, rather than a direct comparison of the entropy level between the two groups with different conditions.

Spike-timing-dependent plasticity (STDP) is a fundamental and biologically plausible concept in the field of neuroscience that has revolutionized our interpretation of synaptic plasticity – the ability of neuronal connections to adapt in response to external stimuli. Unlike traditional models of synaptic plasticity that solely focus on the frequency or intensity of neuronal activity, STDP factors the precise timing of spikes into the mechanism that underlies the crosstalk between neurons. At the heart of it is the asymmetric form of Hebbian learning induced by the correlations between the arrival times of the spikes of pre- and post-synaptic neurons. This phenomenon has profound implications for learning and memory processes in the brain, as it allows synapses to dynamically adjust their efficacy based on the temporal order of neuronal firing [303–307]. As a result, in this work, we will leverage a mathematical model to enable the update of the synaptic weights subject to external stimuli.

In using these mathematical components, we will develop a formalism that allows us to directly investigate the impact of STDP rules as a consequence of the external stimulation on the tendency of entropy change in a reservoir of recurrent neurons. Essentially, we will furnish supporting evidence for the experimental findings mentioned before in a more generic framework and reveal other observations that remain untouched so far in biology and physiology. The chapter will be organized as follows: Section 5.2 presents the models and methods that are used to construct this biologically plausible formalism. Section 5.3 exhibits the simula-

tion results generated by the model. A summary and some further discussions are included in Section 5.4.

## 5.2 Models and methods

### 5.2.1 Overview of the model

The LSMs share a similar structure with ESNs that have been thoroughly introduced in Chapter 4. However, since we are now only concerned with the impact of the input stimuli on the entropy in the reservoir layer through STDP rules, the output layer can be permanently removed and a schematic is shown in Figure 5.1. Note that the input data as well as the neuronal signals in the reservoir layer shall be converted to spike trains using biophysical models for the sake of biological plausibility, as indicated in the schematic diagram. The basic procedure of our ex-

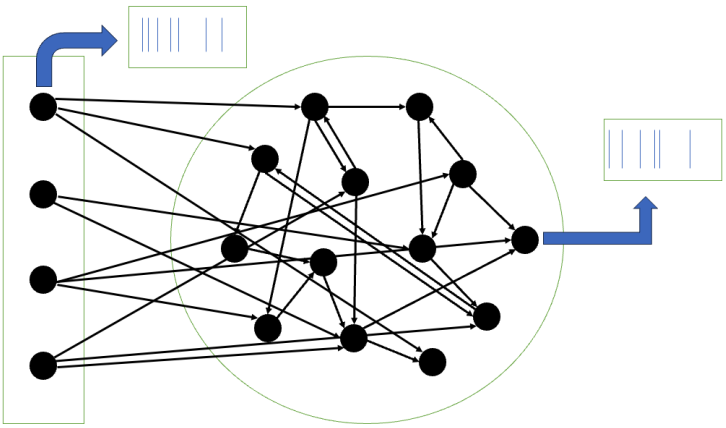


**Figure 5.1:** Liquid state machine (LSM)

periment can be described as follows. We employ the publicly available time-series recordings as the input to the reservoir layer in order to make our simulations better resemble real-world scenarios. To this end, a temporal encoding method will be leveraged to convert the intrinsically continuous sequences into spike trains so as to reduce the loss of temporal information. The neurons inside the reservoir layer will be characterized by the leaky integrate-and-fire (LIF) model by reason of its computational simplicity that can practically facilitate the simulation of large networks. The synaptic weights in the reservoir layer will be adjusted through STDP rules

according to the firing patterns of neurons as a result of the stimulation by the input spike train. Lastly, the entropy values before and after the adaption with respect to the same input sequence will be compared to investigate the effect of external stimulation on the entropy through STDP rules. More details will be provided in the following sections.

### 5.2.2 Temporal encoding of the input sequence

In contrast to artificial neural networks (ANNs) where the original sequence can be directly used as the input to the network, spiking neural networks often require the input sequence to be converted to a spike train as the binary information which is more in agreement with our current understanding of the way that neurons process sensory stimulation. A spike will be generated when the neuron fires at a particular time and nothing otherwise.

From now on, we will restrict our description to one-dimensional sequences for conciseness and later one will easily see how the method can be translated into multivariate cases. Before conducting any conversion procedure, we first introduce the bio-inspired differential operator presented in Section 4.2.2 to be concerned with the contrast in the input:

$$\hat{u}_t = abs(u_{t+1} - u_t), \qquad t = 1, 2, ..., n-1. \tag{5.1}$$

where $(u_t)_{t=1,2,...,n}$ is the representation of a specific sequence and $t$ denotes the timestamp index. Note here we also take the absolute value of the difference so as to only differentiate the degree of change such that the change in either direction can be treated equally important. With that being said, we still want to point out that one can proceed without taking the absolute value and it will not affect the qualitative result of this study. Again, we will hereafter drop the hat on the transformed observation for the sake of simplicity. Lastly, a min-max scaler is applied to the sequences (from the same source) to confine the values between 0 and 1:

$$u_t = \frac{u_t - u_{min}}{u_{max} - u_{min}} \tag{5.2}$$

where $u_{min}$ and $u_{max}$ denote the minimum and the maximum within the whole dataset, respectively.

In order to convert a given time-series sequence into a spike train, the simplest and the most intuitive solution is to binarize the entire sequence with a cut-off threshold. Namely, $(I[u_t > thr])_{t=1,2,\ldots,n}$. Here $u_t$ is the value at timestamp $t$, $I$ is an indicator function and $thr$ is the threshold. However, this simple solution comes with an obvious drawback. Now that the values of the observations of the time series have been turned into binary categories, the relative importance among observations in the same category will be lost and this will greatly impact the resolution of the sequences. To alleviate this, one may consider encoding the relative magnitude into the temporal difference of spikes so that the accuracy of the sequence can be largely preserved. Here we adopted the method using the radial basis functions presented in [308]. Assume any data $u_t \in [a,b]$. In our settings, $a = 0$ and $a = 1$ as we applied the min-max scaler to our original data. Then we choose a series of Gaussian functions as of the radial basis functions with the mean $\mu_i$ and the variance $\sigma^2$.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu_i)^2}{2\sigma^2}} \tag{5.3}$$

where $i = 1,\ldots,N_c$, and $\mu_i$ and $\sigma$ are defined as

$$\begin{cases} \mu_i = a + \dfrac{2i-3}{2}\dfrac{b-a}{N_c-2} \\ \sigma = \dfrac{1}{c}\dfrac{b-a}{N_c-2} \end{cases} \tag{5.4}$$

Here $N_c$ is the number of functions and $c$ is a designed parameter. As such, the value in any time series sequence $u_t$ will be translated into $N_c$ channels with respect to their corresponding bases and can be thereafter converted into their surrogate temporal latency from $t = 0$ to $t = N_c - 1$. More precisely, the temporal latency is determined by the relative distance between $f(u_t)$ and $f(\mu_i)$ such that

$$T_i^t = \left[N_c - \frac{N_c f(u_t)}{f(\mu_i)}\right] \tag{5.5}$$

where the bracket is an operator that rounds the value to the nearest integer and $T_i^t$ essentially indicates the firing time in the time window that concerns this particular value $u_t$ in channel $i$. If the computed firing time lies outside the time window $N_c - 1$, there will be no firing that takes place in the time window. Next we will use $N_c = 5$ and $c = 0.8$ as a quick example. The five Gaussian functions are plotted in Figure 5.2. If $u_t = 1$, then the corresponding values of $f(x)$ will be $0.0015, 0.01900, 0.1296, 0.4660, 0.8838$, and consequently, the corresponding $T_i^t$ will be $5,5,4,3,0$ with respect to $i = 1,2,3,4,5$. In other words, within the time window that is dedicated to $u_t$, no spike is generated in channel 1 and 2, spikes are generated at timestamp 4, 3 and 0 in channel 3, 4 and 5. The firing pattern of $u_{t+1}$ can be produced similarly and it will be passed right after the pattern by $u_t$. In

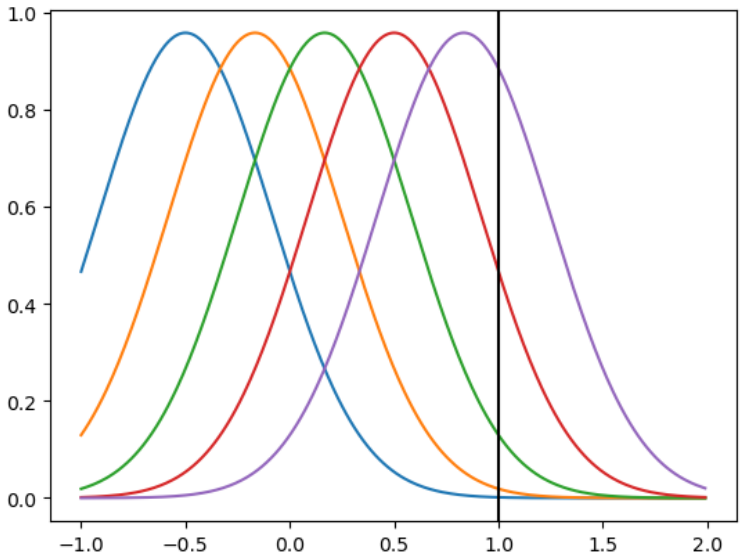



**Figure 5.2:** Illustration of the encoding principle.

summary, if a time series sequence has $k$ features and the length of it is $n$, then the new sequence after the conversion will have $kN_c$ features and the length will be $nN_c$. Here we use $N_c = 10$ and the input sequences are univariate, namely $k = 1$. Lastly, we denote the resulting input spike trains by $\mathbf{S}_t^{in}$.

As one can easily tell, the strength of the method lies in the accurate encoding of continuous data into temporal spikes whereas the main shortcoming is the significantly prolonged sequence which can potentially increase the computational

cost.

### 5.2.3 Neuronal model

As shown in Figure 5.1, the reservoir layer is composed of numerous neurons, the dynamics of which are determined by the input spike trains as well as the interconnections between neurons. In this work, we employ an adapted version of the leaky integrate-and-fire (LIF) model to depict the membrane potentials of the neurons in the reservoir layer. Assume that the reservoir comprises $M$ neurons. Then the membrane potentials of neurons can be denoted by $\mathbf{U}_t \in R^M$ and the spike variables by $\mathbf{S}_t \in \{0,1\}^M$. Accounting for the input signals and the reset of the membrane potentials, the LIF model can be therefore written as [309]

$$\mathbf{U}_t = \beta \mathbf{U}_{t-1} + \mathbf{W}^{in}\mathbf{S}_t^{in} + \mathbf{W}_{t-1}\mathbf{S}_{t-1} - \mathbf{S}_{t-1}\theta \tag{5.6}$$

and

$$\mathbf{S}_t(i) = \begin{cases} 1, & \mathbf{U}_t(i) > \theta \\ 0, & \text{otherwise} \end{cases} \tag{5.7}$$

The derivation of this ODE-free model is shown in Appendix F. Here $\beta$ is the decay rate of the membrane potentials $\mathbf{U}_t$ that dictates the effect of the previous step on the current one. $\mathbf{W}^{in}$ is the matrix of the input connections and remains unchanged throughout the experiment. $\mathbf{W}_t$ is the matrix of the connections in the reservoir layer and will be updated in response to the input signal over time, the mechanism of which will be explained in due course. $\theta$ is the threshold of the action potential and the soft reset is in use to push the membrane potential to the resting state. Here we use $\beta = 0.9$ and $\theta = 0.6$. In Equation 5.7, $i$ indexes the neurons in the reservoir layer and a spike is generated if the membrane potential is above the threshold $\theta$. As compared to the ODE models, the model using discrete update naturally fits the structure of RNNs as the input spike trains can be easily incorporated into the dynamics of neurons. The simplicity of the model also empowers efficient computation when attempting to simulate a large network in order to better epitomize the brain.

Now that we have constructed a biologically plausible model that can be leveraged to coarsely represent the dynamics of neurons in the brain, next we need to further depict the synaptic plasticity to account for the change in synaptic weight in response to external stimulation. The basic STDP model simply computes the difference in the firing times between pre- and post-synaptic neurons and translates it into the synaptic weight:

$$\Delta\omega = E(t_{post} - t_{pre}) \tag{5.8}$$

where $E$ takes the form of

$$E(x) = \begin{cases} A_+ \exp(-\frac{x}{\tau_+}) & x > 0 \\ -A_- \exp(\frac{x}{\tau_-}) & x < 0 \end{cases} \tag{5.9}$$

Here $\Delta\omega$ reflects the change in weight of a synapse. $t_{pre}$ and $t_{post}$ are the firing times of the pre- and post-synaptic neurons. $A_+$, $A_-$, $\tau_+$ and $\tau_-$ are positive parameters. However, the model requires the tracking of the entire history of the pre- and the post-synaptic neurons spikes and the pre- and post-synaptic may fire at the same time step if the step-size is not small enough in a numerical simulation. To this end, in this work, we adopt the online implementation of STDP models for its fine physiological detail and memory efficiency [310]:

$$\begin{cases} \dfrac{d\omega}{dt} = A_+ x_{pre} \sum\limits_i \delta(t - t^i_{post}) - A_- x_{post} \sum\limits_i \delta(t - t^i_{pre}) \\ \tau_+ \dfrac{dx_{pre}}{dt} = -x_{pre} + a_+ \sum\limits_i \delta(t - t^i_{pre}) \\ \tau_- \dfrac{dx_{post}}{dt} = -x_{post} + a_- \sum\limits_i \delta(t - t^i_{post}) \end{cases} \tag{5.10}$$

where $\omega$ is the synaptic weight between two neurons in the reservoir layer, $x_{pre}$ is the pre-synaptic trace variable and $x_{post}$ is the post-synaptic trace variable. There are 6 parameters that control the rate of them, $A_+, A_-, a_+, a_-, \tau_+, \tau_-$ and their contribution can be easily interpreted from the equations. $\delta$ is the delta function. $t^i_{pre}$

and $t^i_{post}$ are the arrival times of the pre- and post-synaptic spikes and $i$ denotes the index of the spikes in the spike train. The pre/post-synaptic spike will elevate the concentration of the trace variable $x_{pre}$/$x_{post}$ and the persistent trace will decay exponentially in the absence of subsequent spikes. The synaptic weight $\omega$ will be substantially increased if the post-synaptic neuron fires at $t^i_{post}$ whilst $x_{pre}$ stays at a high concentration level, whereas $\omega$ will be depressed if the pre-synaptic neuron fires at $t^i_{pre}$ whilst $x_{post}$ stays at a high concentration level. Hence, the change in synaptic weight in the reservoir layer due to the stimulation can be appropriately modelled using the difference in the arrival times of pre- and post-synaptic spikes. The parameter values that we use in this work are $A_+ = A_- = 0.15$, $a_+ = a_- = 0.1$, $\tau_+ = \tau_- = 10$. The time step $\Delta t$ used to simulate the system 5.10 is chosen to be 1, which is in agreement with the assumption made in deriving the Equation 5.6.

Lastly, block entropy will be used to quantify the uncertainty that the reservoir layer possesses. It is the Shannon entropy applied to the time series sequence with $k$ histories of it being considered and it takes the form

$$BE(X_i^{(k)}) = -\sum_j P(X_i^{(k)} = j) \log_2(P(X_i^{(k)} = j)) \tag{5.11}$$

for each neuron $i$. Here $X_i^{(k)}$ denotes the $k$-history variable for neuron $i$ and $j$ denotes all categories that correspond to a k-history block. For instance, $k = 1$ represents that every single timestamp is a realization and reverses back to the basic Shannon entropy and $j \in \{[0],[1]\}$. For $k = 2$, $j \in \{[0,0],[0,1],[1,0],[1,1]\}$. As a result, the total entropy of the neurons in the reservoir layer can be written as

$$BE_{res} = \sum_i BE(X_i^{(k)}) \tag{5.12}$$

### 5.2.4 Experimental design

In this section, we will thoroughly elaborate the procedure of a simple experiment so as to study the impact of STDP rules on the alteration in entropy. Admittedly, there is no such thing as a sensible model for this objective as the mechanism that

underpins the function of the brain remains largely unknown. Nonetheless, a computational model with sufficient physiological details together with a practical procedure can still be implemented which can potentially shed more light on the existing knowledge as well as reveal insightful observations that were not seen before. The steps of our experiment can be summarized as follows:

1. 100 different pairs of matrices $(\mathbf{W}^{in}, \mathbf{W}_0)$ are randomly created.

2. 100 time series sequences are chosen from the same dataset and converted from the continuous values to spike trains.

3. For each pair of the matrices and for each sequence $i$, the corresponding spike train $\mathbf{S}_t^{in}$ is fed into the reservoir. The spiking pattern in the reservoir layer is generated and the total entropy as in Equation 5.12 is calculated and denoted by $E_1^i$. The synaptic weights $\mathbf{W}_t$ remain unchanged at this stage and are still $\mathbf{W}_0$.

4. Feed the spike train $\mathbf{S}_t^{in}$ into the reservoir again but this time the STDP rules as in Equation 5.10 are applied and $\mathbf{W}_t$ is changed accordingly.

5. Now that the synaptic weights have been altered, the spike train $\mathbf{S}_t^{in}$ is again fed into the reservoir and the total entropy is calculated and denoted by $E_2^i$.

6. Reset $\mathbf{W}_t$ to $\mathbf{W}_0$.

7. For each pair of the matrices, $a = \sum_i I(E_1 > E_2)$ is calculated where $I$ is the indicator function.

8. The histogram of $a$ produced by 100 different pairs of matrices is generated.

Admittedly, the experiment setting is far from the precise way that the brain receives and processes the signals. However, as the very first attempt in this line of research, our aim is to directly compare the entropy before and after the impact of the STDP rules when seeing the same input sequence for a number of different random synaptic connections and the experiment presents an intuitive framework. Note that in this work, the initial weight matrix in the reservoir layer $\mathbf{W}_0$ is generated using a

normal distribution $N(0,\sigma^2)$ with $\sigma = 0.2$ and all the diagonal entries are zeroed to avoid the self-loop. The input matrix $\mathbf{W}^{in}$ is first generated by a uniform distribution $U(-0.5,0.5)$ and the entries in the matrix are scaled such that $x = x + 0.5sign(x)$ so as to make sure that the neurons in the reservoir layer can be properly excited. Here $x$ denotes an arbitrary entry in the matrix and $sign(x)$ is used to take the sign of the input $x$. A concise algorithm can be found below.

**Algorithm 1** Algorithm for the experiment

The main objective is to capture for each pair of matrices, how many of 100 time series sequences result in a decrease in entropy after the synaptic weights are adapted by STDP rules in response to external stimulation.

Given 100 different pairs $(\mathbf{W}^{in}, \mathbf{W}_0)$ and 100 time series sequences
**for** each pair of matrices **do**
  $a \leftarrow 0$
  **for** each sequence **do**
    feed $\mathbf{S}_t^{in}$, compute $E_1^i$.
    feed again and $\mathbf{W}_t$ is updated by STDP rules through time.
    feed $\mathbf{S}_t^{in}$, compute $E_2^i$.
    $a \leftarrow a + I(E_1 > E_2)$
  **end for**
**end for**
Plot the histogram of all $a$.

## 5.3 Results

In this section, we will leverage three different sources of time series sequences, one artificial and two real datasets, to delve into the propensity in entropy change of various weight matrices $\mathbf{W}_0$ in response to the external stimuli. The two real datasets that we chose for the experiment are Wafer and ArrowHead from the UCR dataset since their time series sequences are long enough to manifest the STDP effect. The original sequences are converted to spike trains using the method introduced in Section 5.2.2. The artificial data have 10 input channels in order to comply with the input dimension of Wafer and ArrowHead after being converted to spike trains. The length of the data is specified as 1500 and each timestamp is generated by the Bernoulli distribution with the probability of spike occurrence being 0.1.

Figure 5.3 - 5.8 present the histograms of the occurrence of $E_1 > E_2$ among 100 time series sequences with respect to each of 100 different randomly initiated matrices for three datasets and $k = 1, 10$. We will first use the first bar in $M = 20$ in Figure 5.7 as an example to illustrate the function of the figure. The height of the first bar is 20 means that, out of 100 randomly created matrices, the frequency that fewer than 5 sequences out of 100 sequences induce a decline in entropy is 20. In other words, as for these 20 out of 100 matrices, 95% of sequences lead to an increase in entropy as the width of each bar is 5.

Figure 5.3, 5.5, 5.7 are the results for $k = 1$ for three datasets and $k = 1$ is the canonical Shannon entropy as explained previously. For small networks, it is clear that when subject to external stimulation, the vast majority of the weight matrices have the tendency to change in a direction that either increases or decreases the entropy when seeing the same sequence again. One may note that the decline in entropy is the more favourable tendency for the dataset Wafer and ArrowHead (Figure 5.5 and 5.7), not least when the size of the reservoir layer becomes relatively large ($M = 100$ and $M = 200$). However, the tendency is not notable for the artificial data (Figure 5.3) and the increase in entropy seems to be the favourable one for small $M$. This may lead to the conclusion that time dependency in the input sequence can prompt a decrease in entropy for most small networks. When the size of the network further grows, the tendency will largely disappear with $M = 2000$ in Figure 5.5 being an exception.

We will first have a close look into the firing patterns when the entropy is increased or decreased. To this end, the spike trains of two networks with $M = 20$ that lead to a decrease and an increase in entropy have been visualized and are shown in Figure 5.9. As is shown, the decrease in entropy is accompanied by a significant decline in the firing frequency whereas the increase is accompanied by a hike in the firing frequency for some neurons.

In order to shed more light on this difference in distribution between the small and the large networks, we attempt to analyze it through the synaptic weight with the largest change in strength when subject to external stimulation. In Figure 5.10,

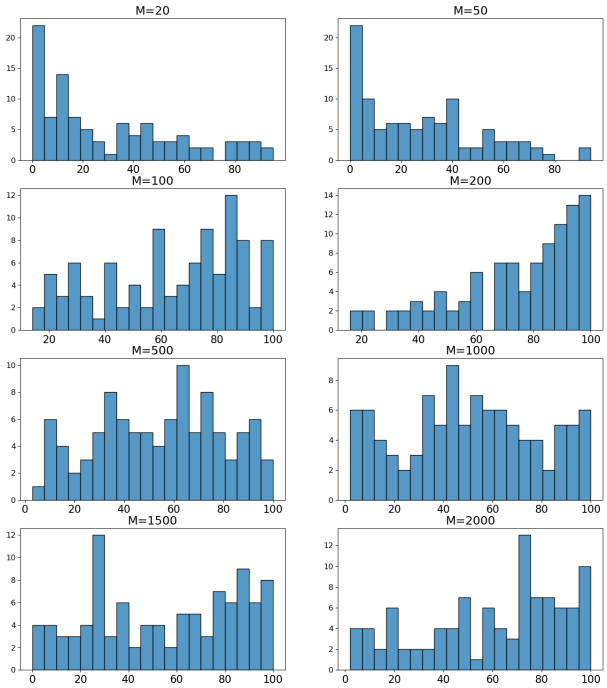



**Figure 5.3:** The histograms of the occurrence of $E_1 > E_2$ among 100 time series sequences with respect to each of 100 different randomly initiated matrices from $M = 20$ to $M = 2000$. The sequences are randomly generated and $k = 1$. The height of each bar in each subfigure indicates the number of pairs of matrices (out of 100) initiated in Algorithm 1. The width of the bar indicates the interval of the cases of decreased entropy out of the 100 time series sequences. The bar essentially summarizes the number of pairs of matrices (out of 100), where a certain number of cases of decreased entropy occurs out of the 100 time series. More details are available in Section 5.2.4 and 5.3.

for $M = 20$, we select a random connection (out of 100) that results in the decrease in entropy for all 100 samples of the dataset Wafer and the trajectories of the weight with the largest change of 20 of them are displayed. As can be seen, there are only two predominant trajectories (straight up or straight down) and the largest variation always happens to the same few synaptic weights with respect to these 20 samples by inspecting the value of the synaptic strength at the start and at the end. Conversely, for $M = 1000$, we select a random connection (out of 100) that results in about 50-50 outcome of increase and decrease in entropy and there is a lot more global and local variability even for those with a similar trend. More importantly, the largest variation happens to a number of synaptic weights, again, by inspecting the value at the start and at the end. As a result, the loss of the tendency in

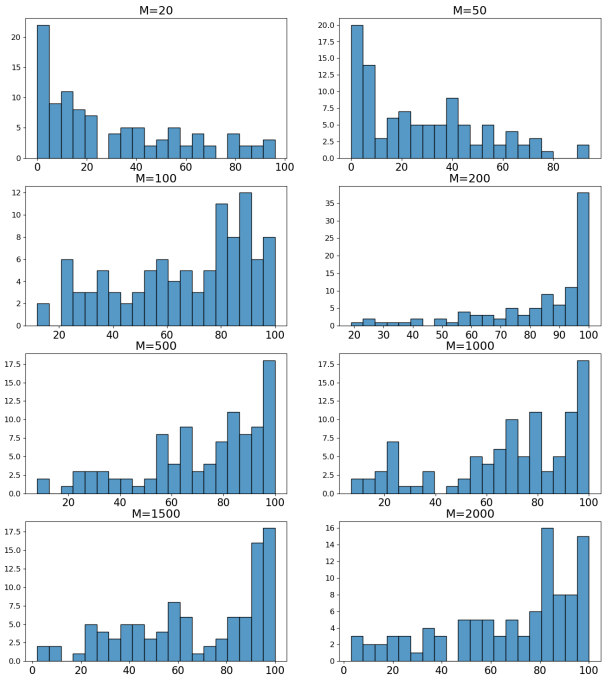



**Figure 5.4:** The histograms of the occurrence of $E_1 > E_2$ among 100 time series sequences with respect to each of 100 different randomly initiated matrices from $M = 20$ to $M = 2000$. The sequences are randomly generated and $k = 10$. The height of each bar in each subfigure indicates the number of pairs of matrices (out of 100) initiated in Algorithm 1. The width of the bar indicates the interval of the cases of decreased entropy out of the 100 time series sequences. The bar essentially summarizes the number of pairs of matrices (out of 100), where a certain number of cases of decreased entropy occurs out of the 100 time series. More details are available in Section 5.2.4 and 5.3.

entropy change in large networks for $k = 1$ may be attributed to the more complex spatial and temporal variation within it such that the change in synaptic weights and therefore, in entropy, becomes less unidirectional.

The next question that we attempt to answer is why the increase in entropy stops being a preferred propensity when the network grows larger, as is reflected from $M = 20, 50$ to $M = 100$ and beyond. Even though it is not possible to provide solid reasoning for this question, one can still try to gain more information from the statistical perspective. To this end, in the case of $M = 20$ and $M = 50$, when the propensity of entropy increase still prevails (sometimes in conjunction with the decreasing propensity), we each choose one initial connection that results in the increase in entropy for all 100 samples, and one that results in decrease for all 100 samples. For each of them, we compute the absolute change in entropy for these

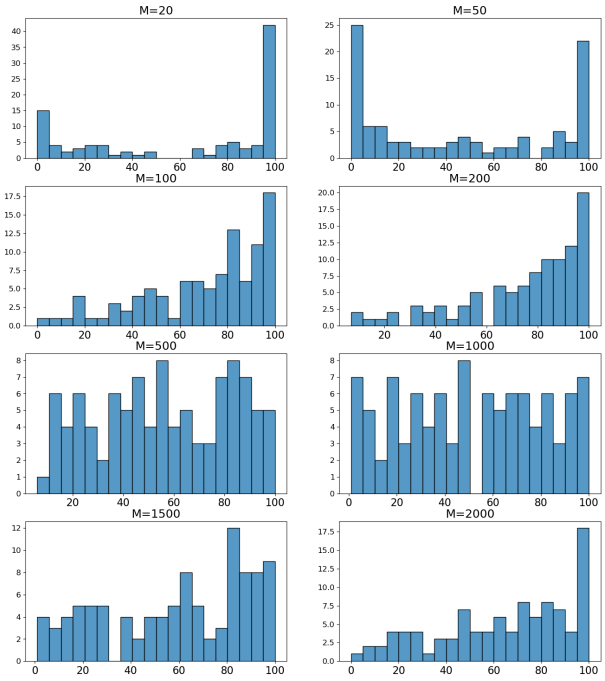



**Figure 5.5:** The histograms of the occurrence of $E_1 > E_2$ among 100 time series sequences with respect to each of 100 different randomly initiated matrices from $M = 20$ to $M = 2000$. The sequences are from the dataset Wafer and $k = 1$. The height of each bar in each subfigure indicates the number of pairs of matrices (out of 100) initiated in Algorithm 1. The width of the bar indicates the interval of the cases of decreased entropy out of the 100 time series sequences. The bar essentially summarizes the number of pairs of matrices (out of 100), where a certain number of cases of decreased entropy occurs out of the 100 time series. More details are available in Section 5.2.4 and 5.3.

100 samples and plot the histogram as shown in Figure 5.11. In each figure, the histogram of the decrease is shown in blue and the increase in orange. As one can observe, the decrease trend seems to have a larger absolute entropy change (not least for $M = 50$) and this may imply that the decrease is more predominant even in the prevalence of the increase trend for many different initial connections.

Figure 5.4, 5.6, 5.8 are the results for $k = 10$ for three datasets. Recall the definition of the Shannon entropy (Equation 5.11). The case $k = 1$ simply explores the binary outcome at each timestamp in the sequence and the entropy is a reflection of the probability of occurrence of each. By contrast, the case $k = 10$ looks into a block of 10 consecutive timestamps and therefore, many more outcomes ($2^{10}$) will be considered accordingly. It can measure the amount of uncertainty in a system at a higher complexity level as the dependency in a short time window has been taken

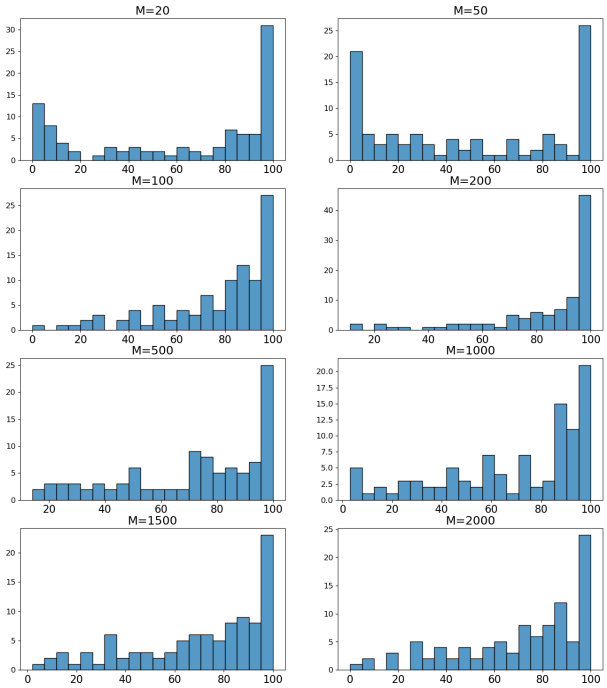



**Figure 5.6:** The histograms of the occurrence of $E_1 > E_2$ among 100 time series sequences with respect to each of 100 different randomly initiated matrices from $M = 20$ to $M = 2000$. The sequences are from the dataset Wafer and $k = 10$. The height of each bar in each subfigure indicates the number of pairs of matrices (out of 100) initiated in Algorithm 1. The width of the bar indicates the interval of the cases of decreased entropy out of the 100 time series sequences. The bar essentially summarizes the number of pairs of matrices (out of 100), where a certain number of cases of decreased entropy occurs out of the 100 time series. More details are available in Section 5.2.4 and 5.3.

into consideration. The distribution of $k = 10$ shares some similarity with $k = 1$ up to a medium-sized network ($M = 200$), apart from the fact that the decrease tendency seems to be more categorical. The distribution of large networks parts with $k = 1$ and the decrease in entropy still remains the overriding tendency, even though the tendency has become less significant. This may suggest the existence of a favored tendency in entropy when the block outcome is considered and provide an explanation for some experimental results in relation to the entropy change subject to external stimulation. However, the networks larger than 2000 neurons are not considered in this work due to the limits of computation.

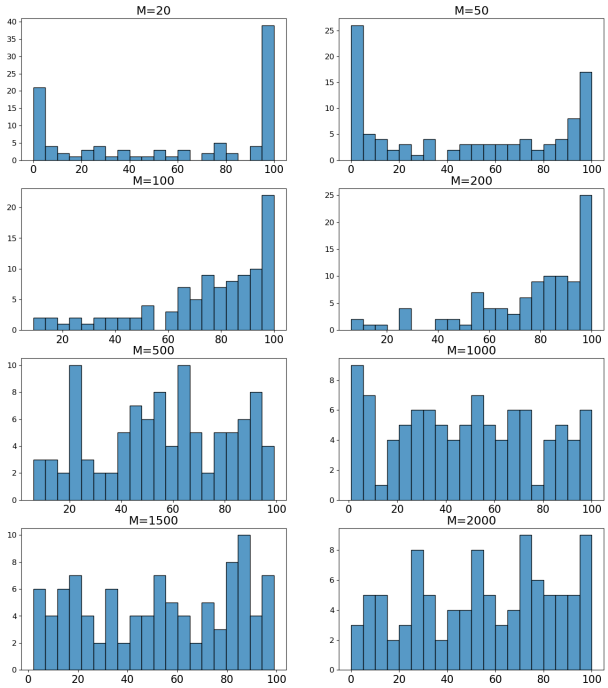



**Figure 5.7:** The histograms of the occurrence of $E_1 > E_2$ among 100 time series sequences with respect to each of 100 different randomly initiated matrices from $M = 20$ to $M = 2000$. The sequences are from the dataset ArrowHead and $k = 1$. The height of each bar in each subfigure indicates the number of pairs of matrices (out of 100) initiated in Algorithm 1. The width of the bar indicates the interval of the cases of decreased entropy out of the 100 time series sequences. The bar essentially summarizes the number of pairs of matrices (out of 100), where a certain number of cases of decreased entropy occurs out of the 100 time series. More details are available in Section 5.2.4 and 5.3.

## 5.4 Conclusion and discussion

In this work, we proposed a biologically plausible framework using biophysical models to investigate the impact of external stimulation on the entropy propensity through STDP rules in a reservoir of neurons, which can be seen as a brain in miniature. Above all, we managed to provide some additional evidence for the experimental findings reported in [90,91] from a modelling viewpoint. Figure 5.4, 5.6 and 5.8 demonstrate that a decrease in entropy ($k = 10$) is the tendency for the majority of relatively large networks and it is also mostly the case for the medium-sized networks when $k = 1$. In addition to this, we also observed a shift in the distribution of the tendency of entropy change when the network grows larger. To the best of our knowledge, results of this kind are still non-existent either in silico or in vivo

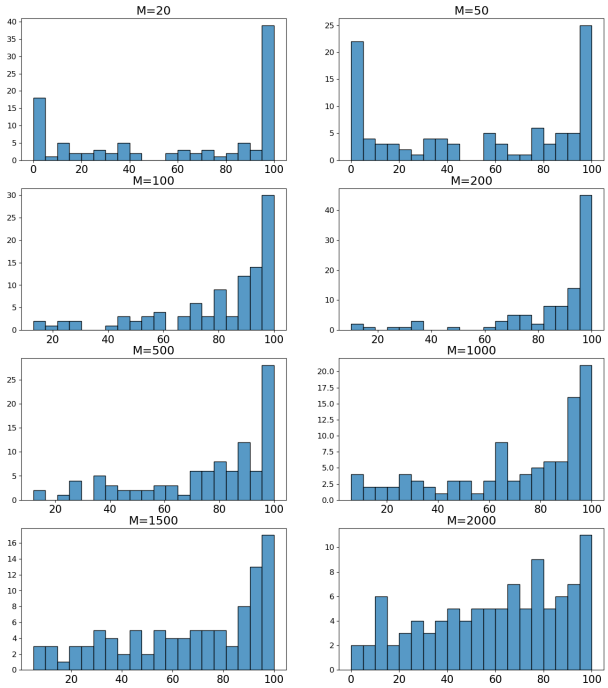
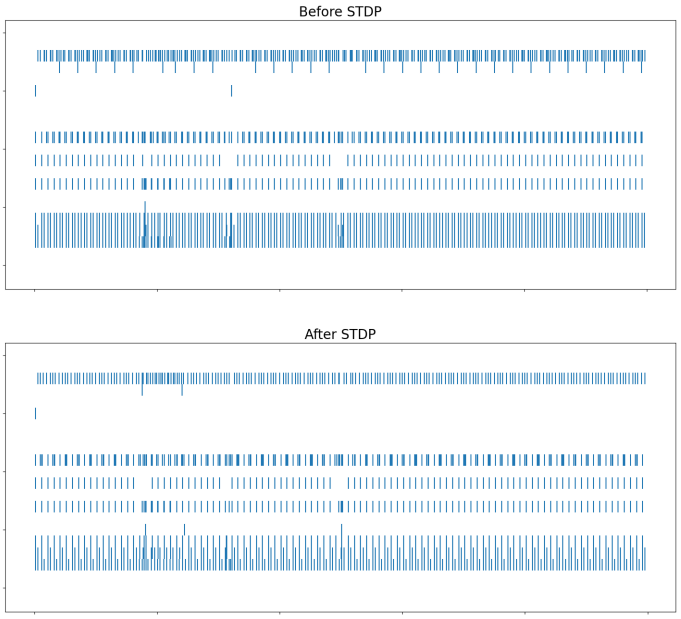


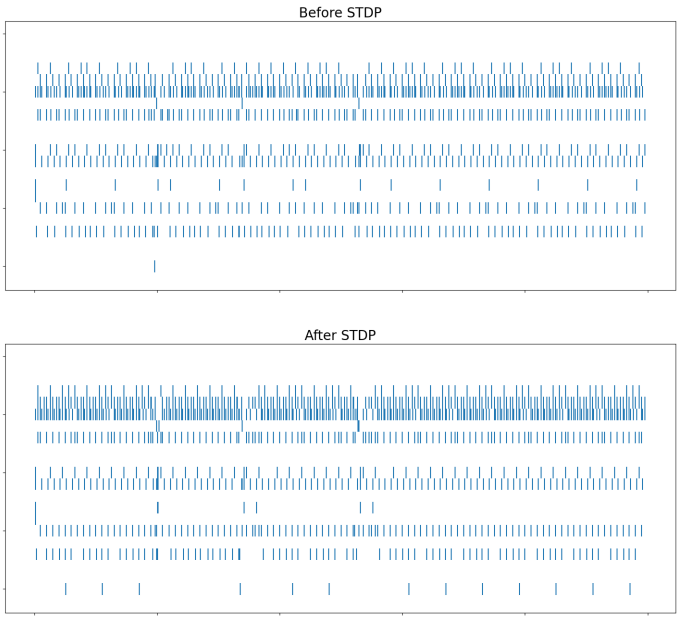

**Figure 5.8:** The histograms of the occurrence of $E_1 > E_2$ among 100 time series sequences with respect to each of 100 different randomly initiated matrices from $M = 20$ to $M = 2000$. The sequences are from the dataset ArrowHead and $k = 10$. The height of each bar in each subfigure indicates the number of pairs of matrices (out of 100) initiated in Algorithm 1. The width of the bar indicates the interval of the cases of decreased entropy out of the 100 time series sequences. The bar essentially summarizes the number of pairs of matrices (out of 100), where a certain number of cases of decreased entropy occurs out of the 100 time series. More details are available in Section 5.2.4 and 5.3.

and our findings might serve as a starting point for the research in this direction. It is possible that a small number of neurons function as a special module and their dynamics are distinct from the global ones. In the meantime, some elucidation has been made in order to better interpret the numerical results that we observed. Last but not least, the connection between the STDP rules and the tendency of entropy change is not trivial. STDP rules can be seen as a type of unsupervised learning. Multiple studies have already employed entropy as an objective function to perform semi-supervised learning [311, 312] and our investigation could open new avenues for the study of the impact of learning on entropy.

Admittedly, our model still has many limitations in terms of biological precision and some further analysis can be carried out. For instance, it may not be



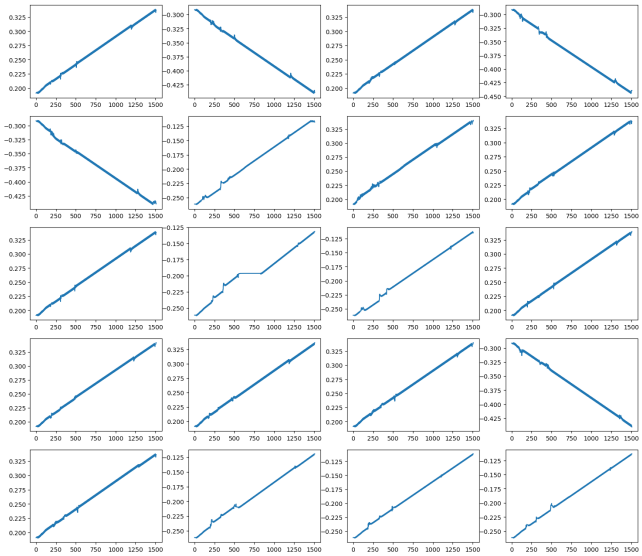

(a) Decrease in entropy



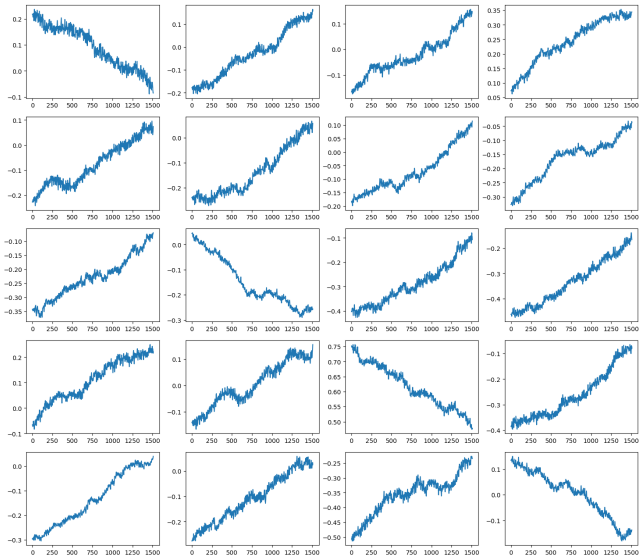

(b) Increase in entropy

**Figure 5.9:** Each figure contains representative spike trains of the neurons in the reservoir before and after the adaption by the STDP rules. Here we use $M = 20$ and we only show the first 1000 time steps for the dataset Wafer.

plausible to assume that under the same external stimulation, the synaptic weights are altered in some process (step 4 in Section 5.2.4) but not in others (step 3 and 5). With that being said, our setting is still arguably the most direct way of illustrating



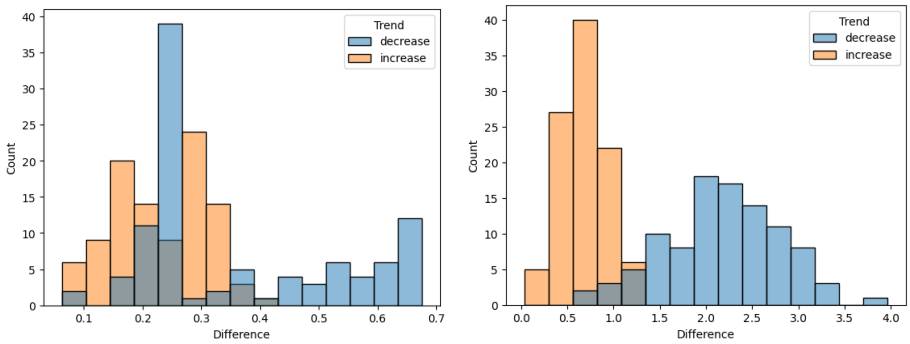
(a) $M = 20$



(b) $M = 1000$

**Figure 5.10:** There are 20 sub-figures in figure (a) and (b). Each sub-figure displays the trajectories of the synaptic weight with the largest change for a specific set of random connections in the input and the reservoir layer ($\mathbf{W}^{in}$ and $\mathbf{W}_0$). The dataset that we used here is Wafer.

the effect of STDP rules as a result of the external stimulation and can be seen as a benchmark for any downstream analysis. Another noticeable limitation is that the entropy calculated thus far is the summation of entropy from individual neurons in the network. Namely, the entropy of concern is computed on the basis of the univariate data and neurons are treated independent during the calculation. It would be




(a) $M = 20$ (b) $M = 50$

**Figure 5.11:** In each figure, we select one initial connection that gives rise to a decrease in entropy for all 100 sequences in the dataset Wafer, and another that gives rise to an increase in entropy for all 100 sequences. Then we compute the absolute value of the change in entropy and plot the histogram for each of them.

more intriguing to investigate the multivariate case where all neurons are considered simultaneously. Moreover, transfer entropy is a measurement of the amount of information transferred between two processes and can be used to better study the information transmission from the input to the reservoir layer [313–316]. As such, a more global perspective on the diversity in firing patterns can be obtained and it can potentially shed more light on the transmission efficiency of the input signal. Nevertheless, computing the entropy of this kind is impossible when the network grows big and a possible solution will be briefly discussed later.

One future direction is to experiment with a more biologically plausible setting and probe into the entropy with and without external stimulation in a single process. One can first let the network evolve when only subject to the background noise and later apply the external current to the network. As such, a parallel can be drawn between the entropy in the early and in the late stage. However, one may need to be careful with the design as we do not want to see an increase in entropy solely as a result of the increasing firing rate.

One can also consider introducing the heterogeneity into the network in order to better resemble the real brain in using the macroscopic description of the neuronal network such as mean-field models [317–320]. At present, all neurons share the same parameters and it may not be a good representation of the neuronal pop-

ulation in the brain. The macroscopic description can be used to characterize the average property (e.g., firing rate, membrane potential) of the neurons in the same population and it enables us to simulate different populations at the same time with lower computational overheads but at the cost of coarse detail. Besides, the mean-field models may also be used to theoretically describe the entropy change in the limit of an infinitely large network, but it is not clear yet how the entropy equation can be incorporated into the description. Lastly, it will also enable the analysis of the aforementioned entropy in the multivariate case as the network can be reduced to a low-dimensional system.

# Chapter 6

# Conclusive summary

In this thesis, we elaborated the new findings about the emerging intelligence within biological systems from the systems biology regard, as a direct result of the spatial and temporal variation among the components in the system, facilitated by the mathematical and statistical models.

In Chapter 2, we revisited the formation of associative learning in the context of genetic circuit and opened the door for the appearance of some advanced properties such as reinforcement effect and forced dissociation. The canonical Fernando's model is capable of exhibiting the reinforcement effect but its formulation is not compatible with the occurrence of the forced dissociation. In order to overcome this limitation, we constructed a new model that can enable reinforcement effect as well as forced dissociation, at the cost of some other inevitable restrictions. Numerical simulations were conducted to predict the outcome of the chemical reactions and the model robustness was also discussed alongside.

In Chapter 3, we attempted to preliminarily address a pioneering but yet largely unanswered question about the impact of astrocytic coverage of synapses on the short-term memory of the visual system together with spatial frequencies. To this end, a newly developed neuron-astrocyte network was adopted to represent the system in miniature and elucidate the change in response. It is facilitated by the interplay among presynaptic neuron, postsynaptic neuron and their perisynaptic astrocytes. In the meantime, we also raised some other hypotheses that may be of interest to the wider community.

In Chapter 4, we founded a new ESN framework that is able to perform TSC tasks with exceptionally low computational overhead. Through the coding of the input sequence into the temporal difference between the adjacent time stamps, the Diff-ESN method succeeds in learning and amplifying the input features in the reservoir layer, thanks to the spatial representation enabled by the intricate recurrent connections. We showed that the method achieved comparable performance with the 1NN-DTW method on a number of benchmark datasets with regular time series. In order to address the more challenging irregular time series, we developed the Interp-ESN method and we demonstrated that it can attain desirable performance on a recent cancer dataset. Moreover, the method also provides the users with an option to predict the outcome in advance.

In Chapter 5, a biologically plausible framework was constructed to represent the means in which the neurons in the brain respond to external stimulation. In specific, we employed the model to study the impact of external stimulation on the tendency in entropy change, the simulation results of which supply some complementary evidence for some experimental observations. Additionally, the shifting pattern of the tendency in entropy change was inspected as the reservoir layer alters in size, opening up an interestingly new research direction for biologists to delve into the effect of network size on information transmission. The temporal and spatial change in the spiking pattern is empowered by a synaptic plasticity rule, STDP.

As shown in the previous chapters, mathematical and statistical models can be used to describe and investigate miscellaneous types of intelligence that some simple biological structures can potentially realize, including associative learning, short-term memory, classification and information processing, etc. Admittedly, the respective subjects were modelled with different levels of biological plausibility, largely attributed to the gap between the theoretical description and the reality. Nonetheless, given that the vast majority of biological systems remain poorly understood, theoretical and numerical analysis is part and parcel when it comes to enriching the existing comprehension as well as proposing plausible hypotheses in order to pave the way for future research. Chapter 2 not only expanded the scope of

the topic based on the current framework, but also pointed to the future engineering of genetic circuits for medical treatment, etc. Indeed, dose control is crucial for the success of treatment and the properties covered in Chapter 2 may make a big difference if appropriately accounted for in vitro and in vivo. For example, the reinforcement effect can indirectly amplify the response concentration and forced dissociation can reduce the level of response in a short time window. In Chapter 3, even though the neuron-astrocyte network simulated was not comparable to the brain on any level, in terms of scale and complexity, it shed light on the connections among the essential components involved in processing visual stimuli and memory storage for the first time. Moreover, one should be aware of the ongoing study of the functional brain where the interactions of various regions are emphasized and one can think of our toy model as a specific region of interest. With the ever-increasing computational power, it may become tractable to bring the model even closer to biological reality. For instance, one can increase the size of the network while creating a modular structure, with convolution-like connections between the input pattern and the network. At first sight, the model employed in Chapter 4 seems quite distant from the way that biological neurons process signals in reality. However, simply binarizing the activation values in the reservoir layer will closely resemble a neuronal circuit in the human brain. One important direction is to continue with the development of neuromorphic computing, for the sake of enabling next-generation computing and a better understanding of the computations carried out in neuronal circuits. Another direction is to improve the performance by introducing light training algorithms to the input and reservoir layer so as to strike a better balance between accuracy and training efficiency. In Chapter 5, the simulation of a relatively large neuronal network with STDP rules has been made possible, thanks to the adoption of the LIF model and parallel computing. In addition to supplying supportive evidence for the existing experimental observations, it also raised the discussion on the entropy change concerning the size of the network and the complexity of the dynamics considered, when subject to external stimulation. Mathematical models also possess a lot more freedom over the settings of the experiments as compared to

the experiments in practice. Hence, the biological detail can be further enhanced by deriving population models enabled by the mean-field description. Besides, some other more advanced measures of information can be accounted for once the computing requirement is met. For example, integrated information may be considered as it measures the amount of information lost if a system is divided into parts. It is reckoned to be overlaid with the functional structure of the brain.

# Appendix A

# Derivation of the translation equation and biological background

In molecular biology, a promoter is a short sequence of DNA upstream of the gene that it regulates. Promoters are regulated by transcription factors (TFs) and within them contains sequences of DNA, known as operator sites that TFs can bind to. Broadly speaking, depending on the type of promoters and TFs, the TFs either activate or inhibit the transcription of the gene after the operator sites are bound by the TFs.

The monomers of a TF typically forms a polymer before binding to the operator site and the biochemical reactions can be expressed as

$$\begin{cases} T + \cdots + T \underset{k_{-p}}{\overset{k_p}{\rightleftharpoons}} T_n \\ O_u + T_n \underset{k_{-b}}{\overset{k_b}{\rightleftharpoons}} O_b \end{cases} \tag{A.1}$$

where $T$ is the concentration of the monomer of the TF, $T_n$ is the concentration of the polymer of order $n$ (or $n$-mer), $O_u$ and $O_b$ are the concentrations of the unbound and the bound operator, respectively. $k_p$, $k_{-p}$, $k_b$, $k_{-b}$ stand for the respective forward and backward reaction rates. Using the law of mass action, the biochemical reactions can therefore be translated into the following equations when the equilibrium

is attained:

$$\begin{cases} k_p T^n = k_{-p} T_n \\ k_b O_u T_n = k_{-b} O_b \end{cases} \tag{A.2}$$

If we assume that the total concentration of DNA is conserved such that

$$O_u + O_b = N \tag{A.3}$$

Hence,

$$\begin{cases} O_u = \dfrac{NK^n}{K^n + T^n} \\ O_b = \dfrac{NT^n}{K^n + T^n} \end{cases} \tag{A.4}$$

where $K = \left(\frac{k_{-p}k_{-b}}{k_p k_b}\right)^{\frac{1}{2}}$. If one moves $N$ to the left and retain the names of the variables, it becomes

$$\begin{cases} O_u = \dfrac{K^n}{K^n + T^n} \\ O_b = \dfrac{T^n}{K^n + T^n} \end{cases} \tag{A.5}$$

Here $K$ is called Hill constant and $n$ is called Hill exponent. Then the rate of transcription can be assumed to have the form

$$r(T) = a_u O_u + a_b O_b \tag{A.6}$$

where $a_u$ and $a_b$ are the parameters that control the contribution of $O_u$ and $O_b$ to the rate. If the TF activates the transcription, we generally assume $a_u = 0$ and likewise, $a_b = 0$ is the TF inhibits the transcription. Lastly, if we make a further assumption that the translation rate (the rate of protein production) is proportional to the rate of transcription, then the translation rate can be written as

$$\frac{dx}{dt} = a_u O_u + a_b O_b - d(x) \tag{A.7}$$

where $x$ is the concentration of the protein and $d(x)$ is the decay rate.

One can also construct the equations for more complex cases in which there are more than one TF involved in a similar way. As a matter of fact, this would

merely be a task of finding the right combination of $O_u$ and $O_b$.

## Appendix B

# Comparing properties of computational neuron models

In Table B.1, we provide a quick overview of whether the biological properties of the firing of cortical neurons can be realized by some of the most widely used biophysical models. The models of concern are the Izhikevich model, the Hodgkin-Huxley model, the FitzHugh-Nagumo model, and the integrate-and-fire model. The table is adapted from [321] and a detailed description of the properties can also be found there. As shown, the Izhikevich model is able to match all the properties that can be simulated from the gold-standard Hodgkin-Huxley model except not being biologically meaningful. The biological meaning has to be inevitably traded for computational efficiency as the contraction of gating variables is necessary.

**Table B.1:** Comparison of the biological properties of Izhikevich (Izh) model, Hodgkin-Huxley (HH) model, FitzHugh-Nagumo (FN) model and integrate-and-fire (IF) model. The first column lists the biological properties in consideration. From the second to the fifth column indicate if the properties can be achieved numerically by the respective models, Yes (Y) or No (N). The last row provides a qualitative description of the computational efficiency of the models.

| **Property** | **Izh** | **HH** | **FN** | **IF** |
|---|---|---|---|---|
| bio-meaningful | N | Y | N | N |
| tonic spiking | Y | Y | Y | Y |
| phasic spiking | Y | Y | Y | N |
| tonic bursting | Y | Y | N | N |
| phasic bursting | Y | Y | N | N |
| mixed mode | Y | Y | Y | N |
| spike frequency adaption | Y | Y | N | N |
| spike latency | Y | Y | Y | N |
| subthreshold oscillation | Y | Y | Y | N |
| chaos | Y | Y | N | N |
| efficiency | excellent | undesirable | good | best |

## Appendix C

# Choosing the scaling factor for the weight matrix in the reservoir layer

**Definition 2.** *The singular values of a matrix* $\mathbf{W}$ *are the non-negative square roots of the eigenvalues of the symmetric positive semi-definite matrix* $\mathbf{W}^T\mathbf{W}$.

**Definition 3.** *The spectral radius of a square matrix* $\mathbf{W}$ *is the maximum of the absolute values of its eigenvalues:* $\rho = \max\{|\lambda_1|, |\lambda_2|, \ldots, |\lambda_n|\}$.

**Proposition 1.** *Assume Equation 4.2 will be used in the reservoir layer to update the internal state and assume* $\mathbf{W}_{res}$ *satisfies* $\sigma_{max} < 1$ *where* $\sigma_{max}$ *is the largest singular value of it. Then* $d(\mathbf{T}(\mathbf{x},\mathbf{u}),\mathbf{T}(\mathbf{x}',\mathbf{u})) < \sigma_{max} d(\mathbf{x},\mathbf{x}')$ *for all inputs* $\mathbf{u}$ *and for all internal states* $\mathbf{x},\mathbf{x}' \in [-1,1]^M$ *and it implies echo state property, where* $d(\mathbf{a},\mathbf{b})$ *is the Euclidean distance between* $\mathbf{a}$ *and* $\mathbf{b}$.

*Proof.* If $\sigma_{max} < 1$

$d(\mathbf{T}(\mathbf{x},\mathbf{u}),\mathbf{T}(\mathbf{x}',\mathbf{u})) \leq d((1-a)\mathbf{x},(1-a)\mathbf{x}') + ad(\mathbf{W}_{res}\mathbf{x},\mathbf{W}_{res}\mathbf{x}') \leq (1-a)d(\mathbf{x},\mathbf{x}') + a\sigma_{max} d(\mathbf{x},\mathbf{x}') = (1-a+a\sigma_{max})d(\mathbf{x},\mathbf{x}') < d(\mathbf{x},\mathbf{x}')$ □

As a result, for any given $\mathbf{W}_{res}$, the scaling $\hat{\mathbf{W}}_{res} = \alpha\mathbf{W}_{res} = \mathbf{W}_{res}/\sigma_{max}$ ensures that the ESP will be satisfied. However, the condition in Proposition 1 is very stringent and as per the investigation in [259,322], the ESP can be largely conserved and the performance can be equally well, if not better, if the scaling factor $\alpha$ is in the marginal vicinity of $1/\rho$. Therefore, we choose $\alpha = 1/\rho$ in this work.

## Appendix D

# Support Vector Machine: binary classification case

For illustrative purposes, here we only confine to the binary classification case using the linear kernel. The objective of the support vector machine algorithm(s) is to find the best hyperplane that separates data into two different classes. To this end, a plane that has the widest margin, namely, the maximum distance between data points from the two classes will be found such that any new unseen data points may be classified with more confidence. Data points lying on the same side of the plane can be assigned to the same class.

The SVM seeks a linear transform on each data point $\mathbf{x}_i$ to minimize the cost function

$$\lambda||\beta||^2 + \sum_i^n (1 - y_i\langle\beta\ ,\ \mathbf{x}_i\rangle)_+ \tag{D.1}$$

where

$$f(x)_+ = max(f(x), 0) \tag{D.2}$$

Here $n$ is the number of data points in the training set, $\lambda$ is a regularization parameter and $\beta$ is the parameter that needs to be optimized. Also in the SVM, the class labels are $\{-1, 1\}$.

An example is shown in Figure D.1. The plane is marked by the solid line and the boundary of the margin is marked by the dash line. The points within the boundary are highlighted by circles.

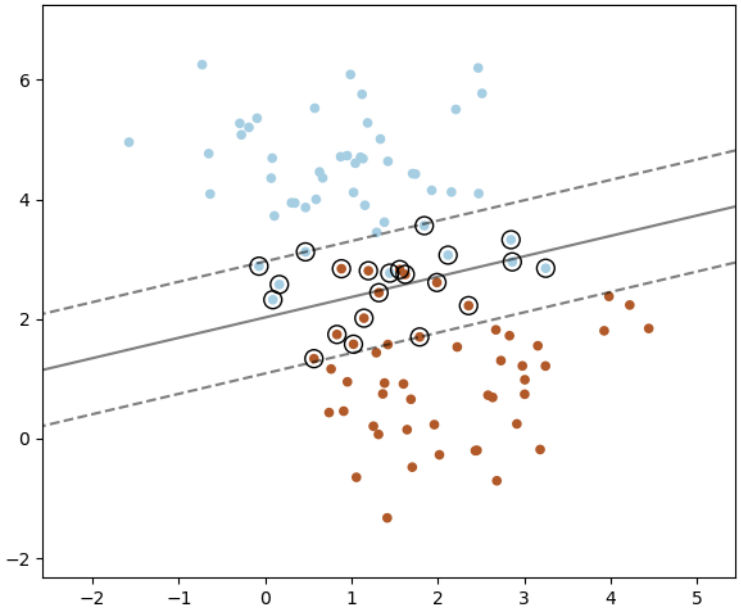



**Figure D.1:** An example of the SVM.

## Appendix E

# Receiver operating characteristic curve (ROC curve)

Firstly, we need to introduce several fundamental definitions that are widely used in binary classification tasks in statistics.

**Definition 4.** ***True Positive (TP)****: the number of cases in which the ground truth class is positive while the prediction is also positive.*

**Definition 5.** ***True Negative (TN)****: the number of cases in which the ground truth class is negative while the prediction is also negative.*

**Definition 6.** ***False Positive (FP)****: the number of cases in which the ground truth class is negative while the prediction is positive.*

**Definition 7.** ***False Negative (FN)****: the number of cases in which the ground truth class is positive while the prediction is negative.*

**Definition 8.** ***True Positive Rate (TPR)*** $= TP / (TP + FN)$

**Definition 9.** ***False Positive Rate (FPR)*** $= FP / (FP + TN)$

A Receiver Operating Characteristic (ROC) curve is a graphical representation that illustrates the performance of a binary classification model at various classification thresholds. It is extensively employed in machine learning and statistics to evaluate the trade-off between sensitivity (TPR) and specificity (1-FPR) of a model

across different threshold values. An ROC curve plots TPR vs. FPR at different classification thresholds. Lowering the classification threshold classifies more items as positive, thus increasing both the TPR and the FPR. An example is shown in Figure E.1. ROC AUC is the area under the ROC curve. ROC AUC = 0.5 rep-

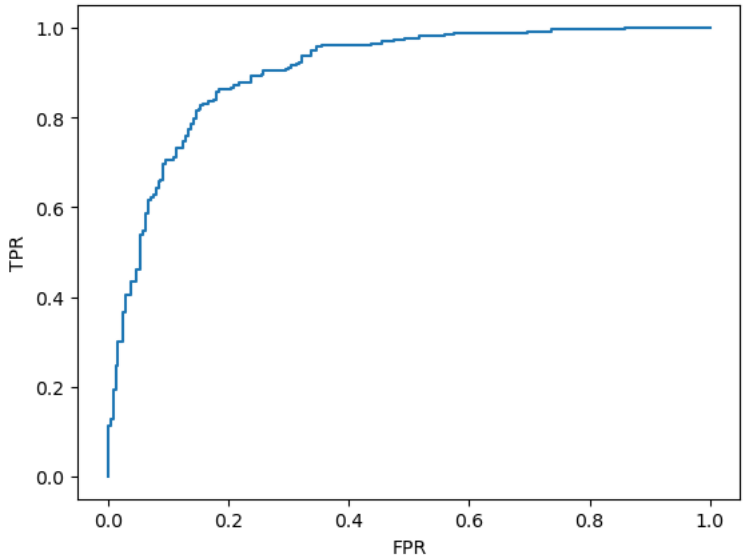



**Figure E.1:** ROC curve

resents that the model has no discrimination power and ROC AUC = 1 represents a model with perfect separation.

## Appendix F

# Derivation of the LIF model shown in Equation 5.6

The dynamics of the passive membrane potential $U$ of a neuron determined by a resistor-capacitor (RC) circuit can be describe as

$$\tau \frac{dU}{dt} = -U + I(t)R \tag{F.1}$$

where $\tau = RC$ is the time constant. Here $C$ is the capacitance and $R$ is the resistance. $I(t)$ is the current being applied to the neuron. Then applying the forward Euler method to Equation F.1 gives rise to

$$\tau \frac{U(t+\Delta t) - U(t)}{\Delta t} = -U(t) + I(t+\Delta t)R \tag{F.2}$$

After some re-arrangement,

$$U(t+\Delta t) = (1 - \frac{\Delta t}{\tau})U(t) + \frac{\Delta t}{\tau} I(t+\Delta t)R \tag{F.3}$$

Next we let $\beta = (1 - \frac{\Delta t}{\tau})$ and further assume that $R = 1\Omega$ and $\Delta t = 1$, it follows that

$$U(t+1) = \beta U(t) + (1-\beta)I(t+1) \tag{F.4}$$

Even though it is not a fully physiologically accurate model as we made the assumption that $R$ and $\Delta t$ should take special values, it somehow makes the model more compatible with the structure of the RNNs which can effectively allows us to handle the update with great ease. Moreover, here $I(t)$ stands for the cumulative current flowing into the neuron and $(1-\beta)$ can be seen as a scaling parameter. Lastly, we make another assumption that the second term can be replaced by the weighted sum of incoming spikes to accommodate the fact that the incoming current into the neuron arrives through the synapses that connect to other presynaptic neurons. Hence, the multi-dimensional version will become

$$\mathbf{U}_{t+1} = \beta \mathbf{U}_t + \mathbf{W}^{in}\mathbf{S}^{in}_{t+1} + \mathbf{W}_t\mathbf{S}_t \tag{F.5}$$

where we moved the time stamp label to the footnote to comply with the notation in the main text. A soft reset term can be added at the end to push the membrane potentials to the resting state:

$$\mathbf{U}_{t+1} = \beta \mathbf{U}_t + \mathbf{W}^{in}\mathbf{S}^{in}_{t+1} + \mathbf{W}_t\mathbf{S}_t - \mathbf{S}_t\theta \tag{F.6}$$

# Bibliography


[1] Melanie Mitchell. Complex systems: Network thinking. *Artificial intelligence*, 170(18):1194–1212, 2006.

[2] Nino Boccara and Nine Boccara. *Modeling complex systems*, volume 1. Springer, 2010.

[3] James Ladyman, James Lambert, and Karoline Wiesner. What is a complex system? *European Journal for Philosophy of Science*, 3:33–67, 2013.

[4] Andrée C Ehresmann and J-P Vanbremeersch. Hierarchical evolutive systems: A mathematical model for complex systems. *Bulletin of Mathematical Biology*, 49(1):13–50, 1987.

[5] Abdelghani Bellouquid and Marcello Delitala. *Mathematical modeling of complex biological systems*. Springer, 2006.

[6] Zhiwei Ji, Ke Yan, Wenyang Li, Haigen Hu, and Xiaoliang Zhu. Mathematical and computational modeling in complex biological systems. *BioMed research international*, 2017(1):5958321, 2017.

[7] Michael K Jensen and Jay D Keasling. Recent applications of synthetic biology tools for yeast metabolic engineering. *FEMS Yeast Res*, 15(1):1–10, 2015.

[8] Tomonori Tamura and Itaru Hamachi. Chemistry for covalent modification of endogenous/native proteins: from test tubes to complex biological systems. *Journal of the American Chemical Society*, 141(7):2782–2799, 2018.

[9] KK Teng and BL Hempstead. Neurotrophins and their receptors: signaling trios in complex biological systems. *Cellular and Molecular Life Sciences CMLS*, 61:35–48, 2004.

[10] Joseph Walpole, Jason A Papin, and Shayn M Peirce. Multiscale computational models of complex biological systems. *Annual review of biomedical engineering*, 15:137–154, 2013.

[11] Rolf Pfeifer and Christian Scheier. *Understanding intelligence*. MIT press, 2001.

[12] Gerardo Beni and Jing Wang. Swarm intelligence in cellular robotic systems. In *Robots and biological systems: towards a new bionics?*, pages 703–712. Springer, 1993.

[13] Satoshi Murata, Taro Toyota, Shin-ichiro M Nomura, Takashi Nakakuki, and Akinori Kuzuya. Molecular cybernetics: challenges toward cellular chemical artificial intelligence. *Advanced Functional Materials*, 32(37):2201866, 2022.

[14] Heather K Titley, Nicolas Brunel, and Christian Hansel. Toward a neurocentric view of learning. *Neuron*, 95(1):19–32, 2017.

[15] Alex JH Fedorec, Neythen J Treloar, Ke Yan Wen, Linda Dekker, Qing Hsuan Ong, Gabija Jurkeviciute, Enbo Lyu, Jack Rutter, Luca Rosa, Alexey Zaikin, et al. Emergent digital bio-computation through spatial diffusion and engineered bacteria. *bioRxiv*, pages 2023–07, 2023.

[16] Steve Furber. Large-scale neuromorphic computing systems. *Journal of neural engineering*, 13(5):051001, 2016.

[17] Ian J Deary, Wendy Johnson, and Lorna M Houlihan. Genetic foundations of human intelligence. *Human genetics*, 126:215–232, 2009.

[18] Kathrin Bohn-Wippert, Erin N Tevonian, Yiyang Lu, Meng-Yao Huang, Melina R Megaridis, and Roy D Dar. Cell size-based decision-making of a viral gene circuit. *Cell reports*, 25(13):3844–3857, 2018.

[19] Geoffrey W Burr, Robert M Shelby, Abu Sebastian, Sangbum Kim, Seyoung Kim, Severin Sidler, Kumar Virwani, Masatoshi Ishii, Pritish Narayanan, Alessandro Fumarola, et al. Neuromorphic computing using non-volatile memory. *Advances in Physics: X*, 2(1):89–124, 2017.

[20] Danijela Marković, Alice Mizrahi, Damien Querlioz, and Julie Grollier. Physics for neuromorphic computing. *Nature Reviews Physics*, 2(9):499–510, 2020.

[21] Mike Davies, Andreas Wild, Garrick Orchard, Yulia Sandamirskaya, Gabriel A Fonseca Guerra, Prasad Joshi, Philipp Plank, and Sumedh R Risbud. Advancing neuromorphic computing with loihi: A survey of results and outlook. *Proceedings of the IEEE*, 109(5):911–934, 2021.

[22] James H Brown, David W Mehlman, and George C Stevens. Spatial variation in abundance. *Ecology*, 76(7):2028–2043, 1995.

[23] J Beuthan, O Minet, J Helfmann, M Herrig, and G Müller. The spatial variation of the refractive index in biological cells. *Physics in Medicine & Biology*, 41(3):369, 1996.

[24] William RL Anderegg. Spatial and temporal variation in plant hydraulic traits and their relevance for climate change impacts on vegetation. *New Phytologist*, 205(3):1008–1014, 2015.

[25] Mia Valtonen, Jukka U Palo, Minna Ruokonen, Mervi Kunnasranta, and Tommi Nyman. Spatial and temporal variation in genetic diversity of an endangered freshwater seal. *Conservation Genetics*, 13:1231–1245, 2012.

[26] Ernesto Andrianantoandro, Subhayu Basu, David K Karig, and Ron Weiss. Synthetic biology: new engineering rules for an emerging discipline. *Molecular systems biology*, 2(1):2006–0028, 2006.

[27] Simon Ausländer, David Ausländer, and Martin Fussenegger. Synthetic biology—the synthesis of biology. *Angewandte Chemie International Edition*, 56(23):6396–6419, 2017.

[28] Mohsen Hesami, Marco Pepe, Austin Baiton, and Andrew Maxwell Phineas Jones. Current status and future prospects in cannabinoid production through in vitro culture and synthetic biology. *Biotechnology Advances*, 62:108074, 2023.

[29] Tzu-Chieh Tang, Bolin An, Yuanyuan Huang, Sangita Vasikaran, Yanyi Wang, Xiaoyu Jiang, Timothy K Lu, and Chao Zhong. Materials design by synthetic biology. *Nature Reviews Materials*, 6(4):332–350, 2021.

[30] Laura Prochazka, Yaakov Benenson, and Peter W Zandstra. Synthetic gene circuits and cellular decision-making in human pluripotent stem cells. *Current Opinion in Systems Biology*, 5:93–103, 2017.

[31] Nuno R Nene, Jordi Garca-Ojalvo, and Alexey Zaikin. Speed-dependent cellular decision making in nonequilibrium genetic circuits. *PloS one*, 7(3):e32779, 2012.

[32] Svetlana Filicheva, Alexey Zaikin, and Oleg Kanakov. Dynamical decision making in a genetic perceptron. *Physica D: Nonlinear Phenomena*, 318:112–115, 2016.

[33] L Abrego and A Zaikin. Decision making in an intracellular genetic classifier. *Mathematical Modelling of Natural Phenomena*, 12(4):30–42, 2017.

[34] Oleg Kanakov, Roman Kotelnikov, Ahmed Alsaedi, Lev Tsimring, Ramon Huerta, Alexey Zaikin, and Mikhail Ivanchenko. Multi-input distributed classifiers for synthetic genetic circuits. *PLoS One*, 10(5):e0125144, 2015.

[35] Javier Macia, Blai Vidiella, and Ricard V Solé. Synthetic associative learning in engineered multicellular consortia. *Journal of The Royal Society Interface*, 14(129):20170158, 2017.

[36] Richard F Thompson, Shaowen Bao, Lu Chen, Benjamin D Cipriano, Jeffrey S Grethe, Jeansok J Kim, Judith K Thompson, Jo Anne Tracy, Martha S Weninger, and David J Krupa. Associative learning. *International review of neurobiology*, 41:151–189, 1997.

[37] Geoffrey Hall and Robert Honey. Perceptual and associative learning. In *Contemporary learning theories*, pages 117–147. Psychology Press, 2014.

[38] Mohamed Adel and Leslie C Griffith. The role of dopamine in associative learning in drosophila: an updated unified model. *Neuroscience bulletin*, 37(6):831–852, 2021.

[39] Tamara Boto, Aaron Stahl, and Seth M Tomchik. Cellular and circuit mechanisms of olfactory associative learning in drosophila. *Journal of neurogenetics*, 34(1):36–46, 2020.

[40] Chrisantha T Fernando, Anthony ML Liekens, Lewis EH Bingle, Christian Beck, Thorsten Lenser, Dov J Stekel, and Jonathan E Rowe. Molecular circuits for associative learning in single-celled organisms. *Journal of the Royal Society Interface*, 6(34):463–469, 2009.

[41] Jan Gründemann and Andreas Lüthi. Ensemble coding in amygdala circuits for associative learning. *Current opinion in neurobiology*, 35:200–206, 2015.

[42] Liqun Luo, Edward M Callaway, and Karel Svoboda. Genetic dissection of neural circuits. *Neuron*, 57(5):634–660, 2008.

[43] Gerald Hahn, Adrian Ponce-Alvarez, Gustavo Deco, Ad Aertsen, and Arvind Kumar. Portraits of communication in neuronal networks. *Nature Reviews Neuroscience*, 20(2):117–127, 2019.

[44] Sonia Cohen and Michael E Greenberg. Communication between the synapse and the nucleus in neuronal development, plasticity, and disease. *Annual review of cell and developmental biology*, 24(1):183–209, 2008.

[45] Zhe Chen, Qian Liang, Zihou Wei, Xie Chen, Qing Shi, Zhiqiang Yu, and Tao Sun. An overview of in vitro biological neural networks for robot intelligence. *Cyborg and Bionic Systems*, 4:0001, 2023.

[46] Ricardo C Cassilhas, Sergio Tufik, and Marco Túlio De Mello. Physical exercise, neuroplasticity, spatial learning and memory. *Cellular and molecular life sciences*, 73:975–983, 2016.

[47] John P McGann. Associative learning and sensory neuroplasticity: how does it happen and what is it good for? *Learning & Memory*, 22(11):567–576, 2015.

[48] Carl CH Petersen and Sylvain Crochet. Synaptic computation and sensory processing in neocortical layer 2/3. *Neuron*, 78(1):28–48, 2013.

[49] Cameron Condylis, Eric Lowet, Jianguang Ni, Karina Bistrong, Timothy Ouellette, Nathaniel Josephs, and Jerry L Chen. Context-dependent sensory processing across primary and secondary somatosensory cortex. *Neuron*, 106(3):515–525, 2020.

[50] Christof Koch. *Biophysics of computation: information processing in single neurons*. Oxford university press, 2004.

[51] Nikolai Axmacher, Florian Mormann, Guillen Fernández, Christian E Elger, and Juergen Fell. Memory formation by neuronal synchronization. *Brain research reviews*, 52(1):170–182, 2006.

[52] Michael J Jutras and Elizabeth A Buffalo. Synchronous neural activity and memory formation. *Current opinion in neurobiology*, 20(2):150–155, 2010.

[53] Vitali Matyash and Helmut Kettenmann. Heterogeneity in astrocyte morphology and physiology. *Brain research reviews*, 63(1-2):2–10, 2010.

[54] Baljit S Khakh and Benjamin Deneen. The emerging nature of astrocyte diversity. *Annual review of neuroscience*, 42:187–207, 2019.

[55] Rafael T Han, Rachel D Kim, Anna V Molofsky, and Shane A Liddelow. Astrocyte-immune cell interactions in physiology and pathology. *Immunity*, 54(2):211–224, 2021.

[56] Michael V Sofroniew. Astrocyte reactivity: subtypes, states, and functions in cns innate immunity. *Trends in immunology*, 41(9):758–770, 2020.

[57] Andrea Volterra, Pierre Julius Magistretti, and Philip G Haydon. *The tripartite synapse: glia in synaptic transmission*. Number BOOK. Oxford University Press, 2002.

[58] Gertrudis Perea, Marta Navarrete, and Alfonso Araque. Tripartite synapses: astrocytes process and control synaptic information. *Trends in neurosciences*, 32(8):421–431, 2009.

[59] Astrocytic Panatier and R Robitaille. Astrocytic mglur5 and the tripartite synapse. *Neuroscience*, 323:29–34, 2016.

[60] Bradley R Postle. Working memory as an emergent property of the mind and brain. *Neuroscience*, 139(1):23–38, 2006.

[61] David EJ Linden. The working memory networks of the human brain. *The Neuroscientist*, 13(3):257–267, 2007.

[62] Yael Shrager, Daniel A Levy, Ramona O Hopkins, and Larry R Squire. Working memory and the organization of brain systems. *Journal of Neuroscience*, 28(18):4818–4822, 2008.

[63] Yuliya Tsybina, Susan Gordleeva, Mikhail Krivonosov, Innokentiy Kastalskiy, Alexey Zaikin, and Alexander Gorban. Modelling working memory in neuron-astrocyte network. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE, 2021.

[64] Susanna Yu Gordleeva, Yuliya A Tsybina, Mikhail I Krivonosov, Mikhail V Ivanchenko, Alexey A Zaikin, Victor B Kazantsev, and Alexander N Gorban. Modeling working memory in a spiking neuron network accompanied by astrocytes. *Frontiers in Cellular Neuroscience*, 15:631485, 2021.

[65] Yuliya Tsybina, Innokentiy Kastalskiy, Mikhail Krivonosov, Alexey Zaikin, Victor Kazantsev, Alexander N Gorban, and Susanna Gordleeva. Astrocytes mediate analogous memory in a multi-layer neuron–astrocyte network. *Neural Computing and Applications*, 34(11):9147–9160, 2022.

[66] Susanna Gordleeva, Yuliya A Tsybina, Mikhail I Krivonosov, Ivan Y Tyukin, Victor B Kazantsev, Alexey Zaikin, and Alexander N Gorban. Situation-based neuromorphic memory in spiking neuron-astrocyte network. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[67] Guy Van der Sande, Daniel Brunner, and Miguel C Soriano. Advances in photonic reservoir computing. *Nanophotonics*, 6(3):561–576, 2017.

[68] Kohei Nakajima. Physical reservoir computing—an introductory perspective. *Japanese Journal of Applied Physics*, 59(6):060501, 2020.

[69] Mushegh Rafayelyan, Jonathan Dong, Yongqi Tan, Florent Krzakala, and Sylvain Gigan. Large-scale optical reservoir computing for spatiotemporal chaotic systems prediction. *Physical Review X*, 10(4):041037, 2020.

[70] Xiangpeng Liang, Jianshi Tang, Yanan Zhong, Bin Gao, He Qian, and Huaqiang Wu. Physical reservoir computing with emerging electronics. *Nature Electronics*, pages 1–14, 2024.

[71] Herbert Jaeger. Echo state network. *scholarpedia*, 2(9):2330, 2007.

[72] Karl Friston. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11(2):127–138, 2010.

[73] Donald W Pfaff. *Brain arousal and information theory: neural and genetic mechanisms*. Harvard University Press, 2006.

[74] Giulio Tononi. Consciousness, information integration, and the brain. *Progress in brain research*, 150:109–126, 2005.

[75] Giulio Tononi. An information integration theory of consciousness. *BMC neuroscience*, 5:1–22, 2004.

[76] Johnjoe McFadden. Integrating information in the brain's em field: the cemi field theory of consciousness. *Neuroscience of consciousness*, 2020(1):niaa016, 2020.

[77] Alfred Wehrl. General properties of entropy. *Reviews of Modern Physics*, 50(2):221, 1978.

[78] Lou Jost. Entropy and diversity. *Oikos*, 113(2):363–375, 2006.

[79] Rocío Martínez Vivot, Carla Pallavicini, Federico Zamberlan, Daniel Vigo, and Enzo Tagliazucchi. Meditation increases the entropy of brain oscillatory activity. *Neuroscience*, 431:40–51, 2020.

[80] David A Drachman. Aging of the brain, entropy, and alzheimer disease. *Neurology*, 67(8):1340–1352, 2006.

[81] Ping Zhou, Paul E Barkhaus, Xu Zhang, and William Zev Rymer. Characterizing the complexity of spontaneous motor unit patterns of amyotrophic lateral sclerosis using approximate entropy. *Journal of Neural Engineering*, 8(6):066010, 2011.

[82] B Paprocki, A Pregowska, and J Szczepanski. Optimizing information processing in brain-inspired neural networks. *Bulletin of the Polish Academy of Sciences. Technical Sciences*, 68(2), 2020.

[83] Guillem Collell and Jordi Fauquet. Brain activity and cognition: a connection from thermodynamics and information theory. *Frontiers in psychology*, 6:145181, 2015.

[84] Mauro Ursino, Giulia Ricci, and Elisa Magosso. Transfer entropy as a measure of brain connectivity: A critical analysis with the help of neural mass models. *Frontiers in computational neuroscience*, 14:45, 2020.

[85] Joel S Perlmutter and Jonathan W Mink. Deep brain stimulation. *Annu. Rev. Neurosci.*, 29:229–257, 2006.

[86] Andres M Lozano, Nir Lipsman, Hagai Bergman, Peter Brown, Stephan Chabardes, Jin Woo Chang, Keith Matthews, Cameron C McIntyre, Thomas E Schlaepfer, Michael Schulder, et al. Deep brain stimulation: current challenges and future directions. *Nature Reviews Neurology*, 15(3):148–160, 2019.

[87] Joachim K Krauss, Nir Lipsman, Tipu Aziz, Alexandre Boutet, Peter Brown, Jin Woo Chang, Benjamin Davidson, Warren M Grill, Marwan I Hariz, Andreas Horn, et al. Technology of deep brain stimulation: current status and future directions. *Nature Reviews Neurology*, 17(2):75–87, 2021.

[88] Youliang Wu, Jiajie Mo, Lisen Sui, Jianguo Zhang, Wenhan Hu, Chao Zhang, Yao Wang, Chang Liu, Baotian Zhao, Xiu Wang, et al. Deep brain stimulation in treatment-resistant depression: a systematic review and meta-analysis on efficacy and safety. *Frontiers in neuroscience*, 15:655412, 2021.

[89] Peggy Wheeler, William E Hoffman, Verna L Baughman, and Heidi Koenig. Response entropy increases during painful stimulation. *Journal of neurosurgical anesthesiology*, 17(2):86–90, 2005.

[90] Alan D Dorval, Gary S Russo, Takao Hashimoto, Weidong Xu, Warren M Grill, and Jerrold L Vitek. Deep brain stimulation reduces neuronal entropy in the mptp-primate model of parkinson's disease. *Journal of neurophysiology*, 100(5):2807–2818, 2008.

[91] Donghui Song, Da Chang, Jian Zhang, Wei Peng, Yuanqi Shang, Xin Gao, and Ze Wang. Reduced brain entropy by repetitive transcranial magnetic

stimulation on the left dorsolateral prefrontal cortex in healthy young adults. *Brain imaging and behavior*, 13:421–429, 2019.

[92] Wolfgang Maass, Thomas Natschläger, and Henry Markram. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural computation*, 14(11):2531–2560, 2002.

[93] Yuta Okumura and Naoki Wakamiya. Analysis of reservoir structure contributing to robustness against structural failure of liquid state machine. In *Artificial Neural Networks and Machine Learning–ICANN 2020: 29th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 15–18, 2020, Proceedings, Part II 29*, pages 435–446. Springer, 2020.

[94] Ning Lin, Shaocong Wang, Yi Li, Bo Wang, Shuhui Shi, Yangu He, Woyu Zhang, Yifei Yu, Yue Zhang, Xiaojuan Qi, et al. Resistive memory-based zero-shot liquid state machine for multimodal event data learning. *arXiv preprint arXiv:2307.00771*, 2023.

[95] Yuniesky Andrade-Talavera, André Fisahn, and Antonio Rodríguez-Moreno. Timing to be precise? an overview of spike timing-dependent plasticity, brain rhythmicity, and glial cells interplay within neuronal circuits. *Molecular Psychiatry*, 28(6):2177–2188, 2023.

[96] Mojtaba Madadi Asl, Abdol-Hossein Vahabie, Alireza Valizadeh, and Peter A Tass. Spike-timing-dependent plasticity mediated by dopamine and its role in parkinson's disease pathophysiology. *Frontiers in Network Physiology*, 2:817524, 2022.

[97] Joel Zirkle and Leonid L Rubchinsky. Spike-timing dependent plasticity effect on the temporal patterning of neural synchronization. *Frontiers in Computational Neuroscience*, 14:52, 2020.

[98] Marius E Yamakou and Christian Kuehn. Combined effects of spike-timing-dependent plasticity and homeostatic structural plasticity on coherence resonance. *Physical Review E*, 107(4):044302, 2023.

[99] Xueyan Hu, Yong Wu, Qianming Ding, Ying Xie, Zhiqiu Ye, and Ya Jia. Synchronization of scale-free neuronal network with small-world property induced by spike-timing-dependent plasticity under time delay. *Physica D: Nonlinear Phenomena*, page 134091, 2024.

[100] Steven A Benner and A Michael Sismour. Synthetic biology. *Nature reviews genetics*, 6(7):533–543, 2005.

[101] D Ewen Cameron, Caleb J Bashor, and James J Collins. A brief history of synthetic biology. *Nature Reviews Microbiology*, 12(5):381–390, 2014.

[102] Ahmad S Khalil and James J Collins. Synthetic biology: applications come of age. *Nature Reviews Genetics*, 11(5):367–379, 2010.

[103] Warren C Ruder, Ting Lu, and James J Collins. Synthetic biology moving into the clinic. *Science*, 333(6047):1248–1252, 2011.

[104] Yensi Flores Bueso, Panos Lehouritis, and Mark Tangney. In situ biomolecule production by bacteria; a synthetic biology approach to medicine. *Journal of Controlled Release*, 275:217–228, 2018.

[105] Ming-Ru Wu, Barbara Jusiak, and Timothy K Lu. Engineering advanced cancer therapies with synthetic biology. *Nature Reviews Cancer*, 19(4):187–195, 2019.

[106] Hugh Douglas Goold, Philip Wright, and Deborah Hailstones. Emerging opportunities for synthetic biology in agriculture. *Genes*, 9(7):341, 2018.

[107] Eleanore T Wurtzel, Claudia E Vickers, Andrew D Hanson, A Harvey Millar, Mark Cooper, Kai P Voss-Fels, Pablo I Nikel, and Tobias J Erb. Revolutionizing agriculture with synthetic biology. *Nature Plants*, 5(12):1207–1210, 2019.

[108] Jing Ke, Bing Wang, and Yasuo Yoshikuni. Microbiome engineering: synthetic biology of plant-associated microbiomes in sustainable agriculture. *Trends in biotechnology*, 39(3):244–261, 2021.

[109] Yensi Flores Bueso and Mark Tangney. Synthetic biology in the driving seat of the bioeconomy. *Trends in biotechnology*, 35(5):373–378, 2017.

[110] J Christopher Anderson, Elizabeth J Clarke, Adam P Arkin, and Christopher A Voigt. Environmentally controlled invasion of cancer cells by engineered bacteria. *Journal of molecular biology*, 355(4):619–627, 2006.

[111] Martin Jinek, Krzysztof Chylinski, Ines Fonfara, Michael Hauer, Jennifer A Doudna, and Emmanuelle Charpentier. A programmable dual-rna–guided dna endonuclease in adaptive bacterial immunity. *science*, 337(6096):816–821, 2012.

[112] Xiaoshu Xu and Lei S Qi. A crispr–dcas toolbox for genetic engineering and synthetic biology. *Journal of molecular biology*, 431(1):34–47, 2019.

[113] Jun Ren, Jingyu Lee, and Dokyun Na. Recent advances in genetic engineering tools based on synthetic biology. *Journal of Microbiology*, 58(1):1–10, 2020.

[114] Randall A Hughes and Andrew D Ellington. Synthetic dna synthesis and assembly: putting the synthetic in synthetic biology. *Cold Spring Harbor perspectives in biology*, 9(1):a023812, 2017.

[115] Yanika Borg, Aurelija Marija Grigonyte, Philipp Boeing, Bethan Wolfenden, Patrick Smith, William Beaufoy, Simon Rose, Tonderai Ratisai, Alexey Zaikin, and Darren N Nesbeth. Open source approaches to establishing roseobacter clade bacteria as synthetic biology chassis for biogeoengineering. *PeerJ*, 4:e2031, 2016.

[116] Darren N Nesbeth, Alexey Zaikin, Yasushi Saka, M Carmen Romano, Claudiu V Giuraniuc, Oleg Kanakov, and Tetyana Laptyeva. Synthetic biol-

ogy routes to bio-artificial intelligence. *Essays in biochemistry*, 60(4):381–391, 2016.

[117] Francesco Bianchini. Artificial intelligence and synthetic biology: a tri-temporal contribution. *Biosystems*, 148:32–39, 2016.

[118] Luis Abrego and Alexey Zaikin. Integrated information as a measure of cognitive processes in coupled genetic repressilators. *Entropy*, 21(4):382, 2019.

[119] Jin Yang, Jihwan Lee, Michelle A Land, Shujuan Lai, Oleg A Igoshin, and François St-Pierre. A synthetic circuit for buffering gene dosage variation between individual mammalian cells. *Nature communications*, 12(1):1–13, 2021.

[120] Jongmin Kim, Kristin S White, and Erik Winfree. Construction of an in vitro bistable circuit from synthetic transcriptional switches. *Molecular systems biology*, 2(1):68, 2006.

[121] William Bacchus, Dominique Aubel, and Martin Fussenegger. Biomedically relevant circuit-design strategies in mammalian synthetic biology. *Molecular systems biology*, 9(1):691, 2013.

[122] Jennifer AN Brophy and Christopher A Voigt. Principles of genetic circuit design. *Nature methods*, 11(5):508–520, 2014.

[123] Hideki Kobayashi, Mads Kaern, Michihiro Araki, Kristy Chung, Timothy S Gardner, Charles R Cantor, and James J Collins. Programmable cells: interfacing natural and engineered gene networks. *Proceedings of the National Academy of Sciences*, 101(22):8414–8419, 2004.

[124] Mingqi Xie and Martin Fussenegger. Designing cell function: assembly of synthetic gene circuits for cell biology applications. *Nature Reviews Molecular Cell Biology*, 19(8):507–525, 2018.

[125] Peng-Fei Xia, Hua Ling, Jee Loon Foo, and Matthew Wook Chang. Synthetic genetic circuits for programmable biological functionalities. *Biotechnology Advances*, 37(6):107393, 2019.

[126] Fabian Tolle, Pascal Stücheli, and Martin Fussenegger. Genetic circuitry for personalized human cell therapy. *Current opinion in biotechnology*, 59:31–38, 2019.

[127] Yanika Borg, Sam Alsford, Vasos Pavlika, Alexei Zaikin, and Darren N Nesbeth. Synthetic biology tools for engineering goodwin oscillation in trypanosoma brucei brucei. *Heliyon*, 8(2):e08891, 2022.

[128] Jesse Stricker, Scott Cookson, Matthew R Bennett, William H Mather, Lev S Tsimring, and Jeff Hasty. A fast, robust and tunable synthetic gene oscillator. *Nature*, 456(7221):516–519, 2008.

[129] Marcel Tigges, Tatiana T Marquez-Lago, Jörg Stelling, and Martin Fussenegger. A tunable synthetic mammalian oscillator. *Nature*, 457(7227):309–312, 2009.

[130] Ekkehard Ullner, Aneta Koseska, Alexey Zaikin, Evgenii Volkov, Jürgen Kurths, and J Garcia-Ojalvo. Dynamics of multicellular synthetic gene networks. *Handbook on Biological Networks, World Scientific, Singapore*, 2009.

[131] Ekkehard Ullner, Alexei Zaikin, Evgenii I Volkov, and Jordi García-Ojalvo. Multistability and clustering in a population of synthetic genetic oscillators via phase-repulsive cell-to-cell communication. *Physical review letters*, 99(14):148103, 2007.

[132] Ekkehard Ullner, Saul Ares, Luis G Morelli, Andrew C Oates, Frank Jülicher, Ernesto Nicola, Raphaela Heussen, David Whitmore, Konstantin Blyuss, Matthew Fryett, et al. Noise and oscillations in biological systems: Multidisciplinary approach between experimental biology, theoretical modelling and synthetic biology. *International Journal of Modern Physics B*, 26(25):1246009, 2012.

[133] Nikolaos Anesiadis, Hideki Kobayashi, William R Cluett, and Radhakrishnan Mahadevan. Analysis and design of a genetic circuit for dynamic metabolic engineering. *ACS synthetic biology*, 2(8):442–452, 2013.

[134] David R Shanks. *The psychology of associative learning.* Cambridge University Press, 1995.

[135] Edward A Wasserman and Ralph R Miller. What's elementary about associative learning? *Annual review of psychology*, 48:573, 1997.

[136] John M Pearce and Mark E Bouton. Theories of associative learning in animals. *Annual review of psychology*, 52(1):111–139, 2001.

[137] Timothy EJ Behrens, Laurence T Hunt, Mark W Woolrich, and Matthew FS Rushworth. Associative learning of social value. *Nature*, 456(7219):245–249, 2008.

[138] Chris J Mitchell, Jan De Houwer, and Peter F Lovibond. The propositional nature of human associative learning. *Behavioral and Brain Sciences*, 32(2):183–198, 2009.

[139] P Ivan Pavlov. Conditioned reflexes: an investigation of the physiological activity of the cerebral cortex. *Annals of neurosciences*, 17(3):136, 2010.

[140] Martin Ziegler, Rohit Soni, Timo Patelczyk, Marina Ignatov, Thorsten Bartsch, Paul Meuffels, and Hermann Kohlstedt. An electronic version of pavlov's dog. *Advanced Functional Materials*, 22(13):2744–2749, 2012.

[141] Michael S Fanselow and Andrew M Poulos. The neuroscience of mammalian associative learning. *Annual review of psychology*, 56(1):207–234, 2005.

[142] Ti Uwano, H Nishijo, T Ono, and R Tamura. Neuronal responsiveness to various sensory stimuli, and associative learning in the rat amygdala. *Neuroscience*, 68(2):339–361, 1995.

[143] Till Faber, Jasdan Joerges, and Randolf Menzel. Associative learning modifies neural representations of odors in the insect brain. *Nature neuroscience*, 2(1):74–78, 1999.

[144] Joshua P Johansen, Lorenzo Diaz-Mataix, Hiroki Hamanaka, Takaaki Ozawa, Edgar Ycu, Jenny Koivumaa, Ashwani Kumar, Mian Hou, Karl Deisseroth, Edward S Boyden, et al. Hebbian and neuromodulatory mechanisms interact to trigger associative memory formation. *Proceedings of the National Academy of Sciences*, 111(51):E5584–E5592, 2014.

[145] Dennis Bray. Molecular networks: the top-down view. *Science*, 301(5641):1864–1865, 2003.

[146] Eshel Ben Jacob, Israela Becker, Yoash Shapira, and Herbert Levine. Bacterial linguistic communication and social intelligence. *TRENDS in Microbiology*, 12(8):366–372, 2004.

[147] Melody K Morris, Julio Saez-Rodriguez, Peter K Sorger, and Douglas A Lauffenburger. Logic-based models for the analysis of cell signaling networks. *Biochemistry*, 49(15):3216–3224, 2010.

[148] Timothy S Gardner, Charles R Cantor, and James J Collins. Construction of a genetic toggle switch in escherichia coli. *Nature*, 403(6767):339–342, 2000.

[149] Piro Siuti, John Yazbek, and Timothy K Lu. Synthetic circuits integrating logic and memory in living cells. *Nature biotechnology*, 31(5):448–452, 2013.

[150] Adrien Padirac, Teruo Fujii, and Yannick Rondelez. Bottom-up construction of in vitro switchable memories. *Proceedings of the National Academy of Sciences*, 109(47):E3212–E3220, 2012.

[151] Baojun Wang, Richard I Kitney, Nicolas Joly, and Martin Buck. Engineering modular and orthogonal genetic logic gates for robust digital-like synthetic biology. *Nature communications*, 2(1):1–9, 2011.

[152] Archibald Vivian Hill. The possible effects of the aggregation of the molecules of haemoglobin on its dissociation curves. *j. physiol.*, 40:4–7, 1910.

[153] Erin L O'Brien, Elizabeth Van Itallie, and Matthew R Bennett. Modeling synthetic gene oscillators. *Mathematical biosciences*, 236(1):1–15, 2012.

[154] Irene Zorzan, Alejandra Rojas López, Anastasiya Malyshava, Tom Ellis, and Matteo Barberis. Synthetic designs regulating cellular transitions: fine-tuning of switches and oscillators. *Current Opinion in Systems Biology*, 25:11–26, 2021.

[155] Michael Fitzgerald, Chelsea Gibbs, Adrian A Shimpi, and Tara L Deans. Adoption of the q transcriptional system for regulating gene expression in stem cells. *ACS synthetic biology*, 6(11):2014–2020, 2017.

[156] Ralph Linsker. Local synaptic learning rules suffice to maximize mutual information in a linear network. *Neural computation*, 4(5):691–702, 1992.

[157] Haifeng Ye, Marie Daoud-El Baba, Ren-Wang Peng, and Martin Fussenegger. A synthetic optogenetic transcription device enhances blood-glucose homeostasis in mice. *Science*, 332(6037):1565–1568, 2011.

[158] Michael Kalos and Carl H June. Adoptive t cell transfer for cancer immunotherapy in the era of synthetic biology. *Immunity*, 39(1):49–60, 2013.

[159] M Krivonosov, T Nazarenko, MG Bacalini, MV Vedunova, C Franceschi, A Zaikin, and M Ivanchenko. Age-related trajectories of dna methylation network markers: a parenclitic network approach to a family-based cohort of patients with down syndrome. *bioRxiv*, pages 2020–03, 2022.

[160] Mikhail Krivonosov, Tatyana Nazarenko, Maria Giulia Bacalini, Alexey Zaikin, Mikhail Ivanchenko, and Claudio Franceschi. Network markers of dna methylation in neurodegenerative diseases. In *2020 4th Scientific School on Dynamics of Complex Networks and their Application in Intellectual Robotics (DCNAIR)*, pages 138–139. IEEE, 2020.

[161] Katrin Rössger, Ghislaine Charpin-El Hamri, and Martin Fussenegger. Reward-based hypertension control by a synthetic brain–dopamine interface. *Proceedings of the National Academy of Sciences*, 110(45):18150–18155, 2013.

[162] Peter E Latham, BJ Richmond, PG Nelson, and S Nirenberg. Intrinsic dynamics in neuronal networks. i. theory. *Journal of neurophysiology*, 83(2):808–827, 2000.

[163] Jaap Van Pelt, Ildiko Vajda, Pieter S Wolters, Michael A Corner, and Ger JA Ramakers. Dynamics and plasticity in developing neuronal networks in vitro. *Progress in brain research*, 147:171–188, 2005.

[164] Alex Roxin and Ernest Montbrió. How effective delays shape oscillatory dynamics in neuronal networks. *Physica D: Nonlinear Phenomena*, 240(3):323–345, 2011.

[165] Hai Zhang, Miaolin Ye, Renyu Ye, and Jinde Cao. Synchronization stability of riemann–liouville fractional delay-coupled complex neural networks. *Physica A: Statistical Mechanics and its Applications*, 508:155–165, 2018.

[166] Christoph M Michel and Thomas Koenig. Eeg microstates as a tool for studying the temporal dynamics of whole-brain neuronal networks: a review. *Neuroimage*, 180:577–593, 2018.

[167] Cendra Agulhon, Jeremy Petravicz, Allison B McMullen, Elizabeth J Sweger, Suzanne K Minton, Sarah R Taves, Kristen B Casper, Todd A Fiacco, and Ken D McCarthy. What is the role of astrocyte calcium in neurophysiology? *Neuron*, 59(6):932–946, 2008.

[168] Heather DE Booth, Warren D Hirst, and Richard Wade-Martins. The role of astrocyte dysfunction in parkinson's disease pathogenesis. *Trends in neurosciences*, 40(6):358–370, 2017.

[169] Raghavendra Upadhya, Winston Zingg, Siddhant Shetty, and Ashok K Shetty. Astrocyte-derived extracellular vesicles: Neuroreparative properties and role in the pathogenesis of neurodegenerative disorders. *Journal of Controlled Release*, 323:225–239, 2020.

[170] Christopher F Rose, Alexei Verkhratsky, and Vladimir Parpura. Astrocyte glutamine synthetase: pivotal in health and disease. *Biochemical Society Transactions*, 41(6):1518–1524, 2013.

[171] Saul W Brusilow, Raymond C Koehler, Richard J Traystman, and Arthur JL Cooper. Astrocyte glutamine synthetase: importance in hyperammonemic syndromes and potential target for therapy. *Neurotherapeutics*, 7(4):452–470, 2010.

[172] Gordleeva Susanna, Kanakov Oleg, Ivanchenko Mikhail, Zaikin Alexey, and Claudio Franceschi. Brain aging and garbage cleaning. In *Seminars in Immunopathology*, volume 42, pages 647–665. Springer Nature BV, 2020.

[173] Philip G Haydon. Glia: listening and talking to the synapse. *Nature Reviews Neuroscience*, 2(3):185–193, 2001.

[174] Vladimir Parpura and Philip G Haydon. Physiological astrocytic calcium levels stimulate glutamate release to modulate adjacent neurons. *Proceedings of the National Academy of Sciences*, 97(15):8629–8634, 2000.

[175] Alfonso Araque, Vladimir Parpura, Rita P Sanzgiri, and Philip G Haydon. Tripartite synapses: glia, the unacknowledged partner. *Trends in neurosciences*, 22(5):208–215, 1999.

[176] Gertrudis Perea and Alfonso Araque. Glia modulates synaptic transmission. *Brain research reviews*, 63(1-2):93–102, 2010.

[177] Alfonso Araque, Vladimir Parpura, Rita P Sanzgiri, and Philip G Haydon. Glutamate-dependent astrocyte modulation of synaptic transmission between cultured hippocampal neurons. *European Journal of Neuroscience*, 10(6):2129–2142, 1998.

[178] Michael M Halassa, Tommaso Fellin, and Philip G Haydon. The tripartite synapse: roles for gliotransmission in health and disease. *Trends in molecular medicine*, 13(2):54–63, 2007.

[179] Mirko Santello, Corrado Calì, and Paola Bezzi. Gliotransmission and the tripartite synapse. *Synaptic Plasticity*, pages 307–331, 2012.

[180] Tommaso Fellin, Olivier Pascual, Sara Gobbo, Tullio Pozzan, Philip G Haydon, and Giorgio Carmignoto. Neuronal synchrony mediated by astrocytic glutamate through activation of extrasynaptic nmda receptors. *Neuron*, 43(5):729–743, 2004.

[181] Paolo Allegrini, Leone Fronzoni, and Davide Pirino. The influence of the astrocyte field on neuronal dynamics and synchronization. *Journal of biological physics*, 35(4):413–423, 2009.

[182] John J Wade, Liam J McDaid, Jim Harkin, Vincenzo Crunelli, and JA Scott Kelso. Bidirectional coupling between astrocytes and neurons mediates learning and dynamic coordination in the brain: a multiple modeling approach. *PloS one*, 6(12):e29445, 2011.

[183] Naiyan Chen, Hiroki Sugihara, Jitendra Sharma, Gertrudis Perea, Jeremy Petravicz, Chuong Le, and Mriganka Sur. Nucleus basalis-enabled stimulus-specific plasticity in the visual cortex is mediated by astrocytes. *Proceedings of the National Academy of Sciences*, 109(41):E2832–E2841, 2012.

[184] S Yu Gordleeva, SA Lebedev, MA Rumyantseva, and Victor B Kazantsev. Astrocyte as a detector of synchronous events of a neural network. *JETP Letters*, 107(7):440–445, 2018.

[185] Evgeniya V Pankratova, Alena I Kalyakulina, Sergey V Stasenko, Susanna Yu Gordleeva, Ivan A Lazarevich, and Viktor B Kazantsev. Neuronal synchronization enhanced by neuron–astrocyte interaction. *Nonlinear Dynamics*, 97(1):647–662, 2019.

[186] Susan Yu Gordleeva, Anastasia V Ermolaeva, Innokentiy A Kastalskiy, and Victor B Kazantsev. Astrocyte as spatiotemporal integrating detector of neuronal activity. *Frontiers in physiology*, 10:294, 2019.

[187] S Yu Makovkin, IV Shkerin, S Yu Gordleeva, and MV Ivanchenko. Astrocyte-induced intermittent synchronization of neurons in a minimal network. *Chaos, Solitons & Fractals*, 138:109951, 2020.

[188] Sergey Makovkin, Evgeny Kozinov, Mikhail Ivanchenko, and Susanna Gordleeva. Controlling synchronization of gamma oscillations by astrocytic modulation in a model hippocampal neural network. *Scientific reports*, 12(1):1–15, 2022.

[189] Yuliya Tsybina, Mikhail Krivonosov, Susan Gordleeva, Alexey Zaikin, and Alexander Gorban. Short-term memory in neuron-astrocyte network. In *2020 4th Scientific School on Dynamics of Complex Networks and their Application in Intellectual Robotics (DCNAIR)*, pages 245–247. IEEE, 2020.

[190] Innokentiy Kastalskiy, Yuliya Tsybina, Victor Kazantsev, and Susanna Gordleeva. Astrocytes' signals guided storage and retrieval of patterns by an snn. In *2021 Third International Conference Neurotechnologies and Neurointerfaces (CNN)*, pages 34–37. IEEE, 2021.

[191] Luis Abrego, Susanna Gordleeva, Oleg Kanakov, Mikhail Krivonosov, and Alexey Zaikin. Estimating integrated information in bidirectional neuron-astrocyte communication. *Physical Review E*, 103(2):022410, 2021.

[192] Oleg Kanakov, Susanna Gordleeva, Anastasia Ermolaeva, Sarika Jalan, and Alexey Zaikin. Astrocyte-induced positive integrated information in neuron-astrocyte ensembles. *Physical Review E*, 99(1):012418, 2019.

[193] Luis Alberto Abrego Rangel. *Information processing in biological complex systems: a view to bacterial and neural complexity*. PhD thesis, UCL (University College London), 2021.

[194] Oleg Kanakov, Susanna Gordleeva, and Alexey Zaikin. Integrated information in the spiking–bursting stochastic model. *Entropy*, 22(12):1334, 2020.

[195] Akinobu Suzuki, Sarah A Stern, Ozlem Bozdagi, George W Huntley, Ruth H Walker, Pierre J Magistretti, and Cristina M Alberini. Astrocyte-neuron lactate transport is required for long-term memory formation. *Cell*, 144(5):810–823, 2011.

[196] Samia Habbas, Mirko Santello, Denise Becker, Hiltrud Stubbe, Giovanna Zappia, Nicolas Liaudet, Federica R Klaus, George Kollias, Adriano Fontana, Christopher R Pryce, et al. Neuroinflammatory tnf$\alpha$ impairs memory via astrocyte signaling. *Cell*, 163(7):1730–1741, 2015.

[197] Adar Adamsky, Adi Kol, Tirzah Kreisel, Adi Doron, Nofar Ozeri-Engelhard, Talia Melcer, Ron Refaeli, Henrike Horn, Limor Regev, Maya Groysman, et al. Astrocytic activation generates de novo neuronal potentiation and memory enhancement. *Cell*, 174(1):59–71, 2018.

[198] Leo Kozachkov and Konstantinos P Michmizos. Sequence learning in associative neuronal-astrocytic networks. In *International Conference on Brain Informatics*, pages 349–360. Springer, 2020.

[199] Simone N De Luca, Alita Soch, Luba Sominsky, Thai-Xinh Nguyen, Abdulhameed Bosakhar, and Sarah J Spencer. Glial remodeling enhances short-term memory performance in wistar rats. *Journal of neuroinflammation*, 17(1):1–18, 2020.

[200] Nelson Cowan. Working memory underpins cognitive development, learning, and education. *Educational psychology review*, 26(2):197–223, 2014.

[201] Alan Baddeley. Working memory: The interface between memory and cognition. *Journal of cognitive neuroscience*, 4(3):281–288, 1992.

[202] James S Nairne. Short-term/working memory. In *Memory*, pages 101–126. Elsevier, 1996.

[203] A Lima, Vanessa Morais Sardinha, AF Oliveira, M Reis, C Mota, MA Silva, Fernanda Marques, JJ Cerqueira, Luisa Pinto, Nuno Sousa, et al. Astrocyte pathology in the prefrontal cortex impairs the cognitive function of rats. *Molecular psychiatry*, 19(7):834–841, 2014.

[204] Laurie M Robin, Jose F Oliveira da Cruz, Valentin C Langlais, Mario Martin-Fernandez, Mathilde Metna-Laurent, Arnau Busquets-Garcia, Luigi Bellocchio, Edgar Soria-Gomez, Thomas Papouin, Marjorie Varilh, et al. Astroglial cb1 receptors determine synaptic d-serine availability to enable recognition memory. *Neuron*, 98(5):935–944, 2018.

[205] Mirko Santello, Nicolas Toni, and Andrea Volterra. Astrocyte function from information processing to cognition and cognitive impairment. *Nature neuroscience*, 22(2):154–166, 2019.

[206] Sonam Akther and Hajime Hirase. Assessment of astrocytes as a mediator of memory and learning in rodents. *Glia*, 70(8):1484–1505, 2022.

[207] Won-Suk Chung, Nicola J Allen, and Cagla Eroglu. Astrocytes control synapse formation, function, and elimination. *Cold Spring Harbor perspectives in biology*, 7(9):a020370, 2015.

[208] Janosch P Heller and Dmitri A Rusakov. Morphological plasticity of astroglia: understanding synaptic microenvironment. *Glia*, 63(12):2133–2151, 2015.

[209] Dmitri A Rusakov. Disentangling calcium-driven astrocyte physiology. *Nature Reviews Neuroscience*, 16(4):226–233, 2015.

[210] Mika Tanaka, Pei-Yu Shih, Hiroshi Gomi, Takamasa Yoshida, Junichi Nakai, Reiko Ando, Teiichi Furuichi, Katsuhiko Mikoshiba, Alexey Semyanov, and Shigeyoshi Itohara. Astrocytic ca2+ signals are required for the functional integrity of tripartite synapses. *Molecular brain*, 6(1):1–13, 2013.

[211] Yu-Wei Wu, Xiaofang Tang, Misa Arizono, Hiroko Bannai, Pei-Yu Shih, Yulia Dembitskaya, Victor Kazantsev, Mika Tanaka, Shigeyoshi Itohara, Katsuhiko Mikoshiba, et al. Spatiotemporal calcium dynamics in single astrocytes and its modulation by neuronal activity. *Cell calcium*, 55(2):119–129, 2014.

[212] Valentina Kustikova, Mikhail Krivonosov, Alexey Pimashkin, Pavel Denisov, Alexey Zaikin, Mikhail Ivanchenko, Iosif Meyerov, and Alexey Semyanov. Calciumcv: Computer vision software for calcium signaling in astrocytes. In *International Conference on Analysis of Images, Social Networks and Texts*, pages 168–179. Springer, 2018.

[213] Valeri Matrosov, Susan Gordleeva, Natalia Boldyreva, Eshel Ben-Jacob, Victor Kazantsev, and Maurizio De Pittà. Emergence of regular and complex calcium oscillations by inositol 1, 4, 5-trisphosphate signaling in astrocytes. In *Computational Glioscience*, pages 151–176. Springer, 2019.

[214] Eugene M Izhikevich. Simple model of spiking neurons. *IEEE Transactions on neural networks*, 14(6):1569–1572, 2003.

[215] VB Kazantsev and S Yu Asatryan. Bistability induces episodic spike communication by inhibitory neurons in neuronal networks. *Physical Review E*, 84(3):031913, 2011.

[216] Victor Kazantsev, Susan Gordleeva, Sergey Stasenko, and Alexander Dityatev. A homeostatic model of neuronal firing governed by feedback signals from the extracellular matrix. 2012.

[217] S Yu Gordleeva, Sergey V Stasenko, Alexey V Semyanov, Alexander E Dityatev, and Victor B Kazantsev. Bi-directional astrocytic regulation of neuronal

activity within a network. *Frontiers in computational neuroscience*, 6:92, 2012.

[218] Michael J Berridge. The inositol trisphosphate/calcium signaling pathway in health and disease. *Physiological reviews*, 96(4):1261–1296, 2016.

[219] Michael J Berridge. Inositol trisphosphate and calcium signalling. *Nature*, 361(6410):315–325, 1993.

[220] Suhita Nadkarni and Peter Jung. Modeling synaptic transmission of the tripartite synapse. *Physical biology*, 4(1):1, 2007.

[221] Franziska Oschmann, Hugues Berry, Klaus Obermayer, and Kerstin Lenk. From in silico astrocyte cell models to neuron-astrocyte network models: A review. *Brain research bulletin*, 136:76–84, 2018.

[222] Langzhou Liu, Huayi Gao, Alexey Zaikin, and Shangbin Chen. Unraveling a$\beta$-mediated multi-pathway calcium dynamics in astrocytes: Implications for alzheimer's disease treatment from simulations. *Frontiers in physiology*, 12, 2021.

[223] Yu-Wei Wu, Susan Gordleeva, Xiaofang Tang, Pei-Yu Shih, Yulia Dembitskaya, and Alexey Semyanov. Morphological profile determines the frequency of spontaneous calcium events in astrocytic processes. *Glia*, 67(2):246–262, 2019.

[224] Ghanim Ullah, Peter Jung, and Ann H Cornell-Bell. Anti-phase calcium oscillations in astrocytes via inositol (1, 4, 5)-trisphosphate regeneration. *Cell calcium*, 39(3):197–208, 2006.

[225] Yuki Fujii, Shohei Maekawa, and Mitsuhiro Morita. Astrocyte calcium waves propagate proximally by gap junction and distally by extracellular diffusion of atp released from volume-regulated anion channels. *Scientific reports*, 7(1):13115, 2017.

[226] Christian Giaume and Laurent Venance. Intercellular calcium signaling and gap junctional communication in astrocytes. *Glia*, 24(1):50–64, 1998.

[227] Mati Goldberg, Maurizio De Pittà, Vladislav Volman, Hugues Berry, and Eshel Ben-Jacob. Nonlinear gap junctions enable long-distance propagation of pulsating calcium waves in astrocyte networks. *PLoS computational biology*, 6(8):e1000909, 2010.

[228] Alexey Semyanov, Christian Henneberger, and Amit Agarwal. Making sense of astrocytic calcium signals—from acquisition to interpretation. *Nature Reviews Neuroscience*, 21(10):551–564, 2020.

[229] Mireille Bélanger, Igor Allaman, and Pierre J Magistretti. Brain energy metabolism: focus on astrocyte-neuron metabolic cooperation. *Cell metabolism*, 14(6):724–738, 2011.

[230] Jillian L Stobart and Christopher M Anderson. Multifunctional role of astrocytes as gatekeepers of neuronal energy supply. *Frontiers in cellular neuroscience*, 7:38, 2013.

[231] Ksenia V Kastanenka, Rubén Moreno-Bote, Maurizio De Pittà, Gertrudis Perea, Abel Eraso-Pichot, Roser Masgrau, Kira E Poskanzer, and Elena Galea. A roadmap to integrate astrocytes into systems neuroscience. *Glia*, 68(1):5–26, 2020.

[232] Amirhossein Tavanaei and Anthony S Maida. Multi-layer unsupervised learning in a spiking convolutional neural network. In *2017 international joint conference on neural networks (IJCNN)*, pages 2023–2030. IEEE, 2017.

[233] Chankyu Lee, Gopalakrishnan Srinivasan, Priyadarshini Panda, and Kaushik Roy. Deep spiking convolutional neural network trained with unsupervised spike-timing-dependent plasticity. *IEEE Transactions on Cognitive and Developmental Systems*, 11(3):384–394, 2018.

[234] Amirhossein Tavanaei, Masoud Ghodrati, Saeed Reza Kheradpisheh, Timothée Masquelier, and Anthony Maida. Deep learning in spiking neural networks. *Neural networks*, 111:47–63, 2019.

[235] Aude Panatier, Dionysia T Theodosis, Jean-Pierre Mothet, Bastien Touquet, Loredano Pollegioni, Dominique A Poulain, and Stéphane HR Oliet. Glia-derived d-serine controls nmda receptor activity and synaptic memory. *Cell*, 125(4):775–784, 2006.

[236] Ulrike Pannasch, Dominik Freche, Glenn Dallérac, Grégory Ghézali, Carole Escartin, Pascal Ezan, Martine Cohen-Salmon, Karim Benchenane, Veronica Abudara, Amandine Dufour, et al. Connexin 30 sets synaptic strength by controlling astroglial synapse invasion. *Nature neuroscience*, 17(4):549–558, 2014.

[237] Elizabeth Ann Maharaj and Andrés M Alonso. Discriminant analysis of multivariate time series: Application to diagnosis based on ecg signals. *Computational Statistics & Data Analysis*, 70:67–87, 2014.

[238] Ziv Bar-Joseph, Anthony Gitter, and Itamar Simon. Studying and modelling dynamic biological processes using time-series gene expression data. *Nature Reviews Genetics*, 13(8):552–564, 2012.

[239] Young Shin Kim, Svetlozar T Rachev, Michele Leonardo Bianchi, Ivan Mitov, and Frank J Fabozzi. Time series analysis for financial market meltdowns. *Journal of Banking & Finance*, 35(8):1879–1891, 2011.

[240] Alex Graves, Santiago Fernández, and Jürgen Schmidhuber. Bidirectional lstm networks for improved phoneme classification and recognition. In *International conference on artificial neural networks*, pages 799–804. Springer, 2005.

[241] Rui Zhao, Ruqiang Yan, Jinjiang Wang, and Kezhi Mao. Learning to monitor machine health with convolutional bi-directional lstm networks. *Sensors*, 17(2):273, 2017.

[242] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A review of recurrent neural networks: Lstm cells and network architectures. *Neural computation*, 31(7):1235–1270, 2019.

[243] Nikita Sushentsev, Leonardo Rundo, Luis Abrego, Zonglun Li, Tatiana Nazarenko, Anne Y Warren, Vincent J Gnanapragasam, Evis Sala, Alexey Zaikin, Tristan Barrett, et al. Time series radiomics for the prediction of prostate cancer progression in patients on active surveillance. *European Radiology*, pages 1–9, 2023.

[244] Amaia Abanda, Usue Mori, and Jose A Lozano. A review on distance based time series classification. *Data Mining and Knowledge Discovery*, 33(2):378–412, 2019.

[245] Jiangyuan Mei, Meizhu Liu, Yuan-Fang Wang, and Huijun Gao. Learning a mahalanobis distance-based dynamic time warping measure for multivariate time series classification. *IEEE transactions on Cybernetics*, 46(6):1363–1374, 2015.

[246] Liuyi Yao, Yaliang Li, Yezheng Li, Hengtong Zhang, Mengdi Huai, Jing Gao, and Aidong Zhang. Dtec: Distance transformation based early time series classification. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 486–494. SIAM, 2019.

[247] Rohit J Kate. Using dynamic time warping distances as features for improved time series classification. *Data Mining and Knowledge Discovery*, 30:283–312, 2016.

[248] Jianyu Long, Shaohui Zhang, and Chuan Li. Evolving deep echo state networks for intelligent fault diagnosis. *IEEE Transactions on Industrial Informatics*, 16(7):4928–4937, 2019.

[249] Vladimir Ivanov and Konstantinos Michmizos. Increasing liquid state machine performance with edge-of-chaos dynamics organized by astrocyte-

modulated plasticity. *Advances in neural information processing systems*, 34:25703–25719, 2021.

[250] Zonglun Li, Alya Fattah, Peter Timashev, and Alexey Zaikin. An account of models of molecular circuits for associative learning with reinforcement effect and forced dissociation. *Sensors*, 22(15):5907, 2022.

[251] Zonglun Li, Yuliya Tsybina, Susanna Gordleeva, and Alexey Zaikin. Impact of astrocytic coverage of synapses on the short-term memory of a computational neuron-astrocyte network. *Mathematics*, 10(18):3275, 2022.

[252] Vladimir Nikolić, Moriah Echlin, Boris Aguilar, and Ilya Shmulevich. Computational capabilities of a multicellular reservoir computing system. *Plos one*, 18(4):e0282122, 2023.

[253] Gouhei Tanaka, Toshiyuki Yamane, Jean Benoit Héroux, Ryosho Nakane, Naoki Kanazawa, Seiji Takeda, Hidetoshi Numata, Daiju Nakano, and Akira Hirose. Recent advances in physical reservoir computing: A review. *Neural Networks*, 115:100–123, 2019.

[254] Kohei Nakajima and Ingo Fischer. *Reservoir Computing*. Springer, 2021.

[255] Daniel J Gauthier, Erik Bollt, Aaron Griffith, and Wendson AS Barbosa. Next generation reservoir computing. *Nature communications*, 12(1):5564, 2021.

[256] Luís F Seoane. Evolutionary aspects of reservoir computing. *Philosophical Transactions of the Royal Society B*, 374(1774):20180377, 2019.

[257] Fabrizio Damicelli, Claus C Hilgetag, and Alexandros Goulas. Brain connectivity meets reservoir computing. *PLoS Computational Biology*, 18(11):e1010639, 2022.

[258] Hongwei Cai, Zheng Ao, Chunhui Tian, Zhuhao Wu, Hongcheng Liu, Jason Tchieu, Mingxia Gu, Ken Mackie, and Feng Guo. Brain organoid reservoir computing for artificial intelligence. *Nature Electronics*, pages 1–8, 2023.

[259] Herbert Jaeger. The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, 148(34):13, 2001.

[260] Taehwan Kim and Brian R King. Time series prediction using deep echo state networks. *Neural Computing and Applications*, 32:17769–17787, 2020.

[261] Huanling Hu, Lin Wang, and Sheng-Xiang Lv. Forecasting energy consumption and wind power generation using deep echo state network. *Renewable Energy*, 154:598–613, 2020.

[262] Herbert Jaeger, Mantas Lukoševičius, Dan Popovici, and Udo Siewert. Optimization and applications of echo state networks with leaky-integrator neurons. *Neural networks*, 20(3):335–352, 2007.

[263] Alexander Borst and Frédéric E Theunissen. Information theory and neural coding. *Nature neuroscience*, 2(11):947–957, 1999.

[264] Rodrigo Quian Quiroga and Stefano Panzeri. *Principles of neural coding*. CRC Press, 2013.

[265] Maoz Shamir. Emerging principles of population coding: in search for the neural code. *Current opinion in neurobiology*, 25:140–148, 2014.

[266] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106, 1962.

[267] Ad MHJ Aertsen and PIM Johannesma. The spectro-temporal receptive field: a functional characteristic of auditory neurons. *Biological cybernetics*, 42(2):133–143, 1981.

[268] Charles Rougé, Yan Ge, and Ximing Cai. Detecting gradual and abrupt changes in hydrological records. *Advances in Water Resources*, 53:33–44, 2013.

[269] Samaneh Aminikhanghahi and Diane J Cook. A survey of methods for time series change point detection. *Knowledge and information systems*, 51(2):339–367, 2017.

[270] Bedartha Goswami, Niklas Boers, Aljoscha Rheinwalt, Norbert Marwan, Jobst Heitzig, Sebastian FM Breitenbach, and Jürgen Kurths. Abrupt transitions in time series with uncertainties. *Nature communications*, 9(1):48, 2018.

[271] Jason Weston and Chris Watkins. Multi-class support vector machines. Technical report, Technical Report CSD-TR-98-04, Department of Computer Science, Royal . . . , 1998.

[272] Donna Badgwell and Robert C Bast Jr. Early detection of ovarian cancer. *Disease markers*, 23(5-6):397–410, 2007.

[273] Ian J Jacobs, Usha Menon, Andy Ryan, Aleksandra Gentry-Maharaj, Matthew Burnell, Jatinderpal K Kalsi, Nazar N Amso, Sophia Apostolidou, Elizabeth Benjamin, Derek Cruickshank, et al. Ovarian cancer screening and mortality in the uk collaborative trial of ovarian cancer screening (ukctocs): a randomised controlled trial. *The Lancet*, 387(10022):945–956, 2016.

[274] Saundra S Buys, Edward Partridge, Amanda Black, Christine C Johnson, Lois Lamerato, Claudine Isaacs, Douglas J Reding, Robert T Greenlee, Lance A Yokochi, Bruce Kessel, et al. Effect of screening on ovarian cancer mortality: the prostate, lung, colorectal and ovarian (plco) cancer screening randomized controlled trial. *Jama*, 305(22):2295–2303, 2011.

[275] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

[276] Hoang Anh Dau, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, and Ea-

monn Keogh. The ucr time series archive. *IEEE/CAA Journal of Automatica Sinica*, 6(6):1293–1305, 2019.

[277] Oleg Blyuss, Alex Gentry-Maharaj, Evangelia-Orania Fourkala, Andy Ryan, Alexey Zaikin, Usha Menon, Ian Jacobs, and John F Timms. Serial patterns of ovarian cancer biomarkers in a prediagnosis longitudinal dataset. *BioMed research international*, 2015, 2015.

[278] Ines P Marino, Oleg Blyuss, Andy Ryan, Aleksandra Gentry-Maharaj, John F Timms, Anne Dawnay, Jatinderpal Kalsi, Ian Jacobs, Usha Menon, and Alexey Zaikin. Change-point of multiple biomarkers in women with ovarian cancer. *Biomedical Signal Processing and Control*, 33:169–177, 2017.

[279] Manuel A Vázquez, Inés P Mariño, Oleg Blyuss, Andy Ryan, Aleksandra Gentry-Maharaj, Jatinderpal Kalsi, Ranjit Manchanda, Ian Jacobs, Usha Menon, and Alexey Zaikin. A quantitative performance study of two automatic methods for the diagnosis of ovarian cancer. *Biomedical Signal Processing and Control*, 46:86–93, 2018.

[280] Parsa Charkhchi, Cezary Cybulski, Jacek Gronwald, Fabian Oliver Wong, Steven A Narod, and Mohammad R Akbari. Ca125 and ovarian cancer: a comprehensive review. *Cancers*, 12(12):3730, 2020.

[281] Minghai Zhang, Shanshan Cheng, Yue Jin, Yaqian Zhao, and Yu Wang. Roles of ca125 in diagnosis, prediction, and oncogenesis of ovarian cancer. *Biochimica et Biophysica Acta (BBA)-Reviews on Cancer*, 1875(2):188503, 2021.

[282] Mehak Khan, Hongzhi Wang, Adnan Riaz, Aya Elfatyany, and Sajida Karim. Bidirectional lstm-rnn-based hybrid deep learning frameworks for univariate time series classification. *The Journal of Supercomputing*, 77:7021–7045, 2021.

[283] Parami Wijesinghe, Gopalakrishnan Srinivasan, Priyadarshini Panda, and Kaushik Roy. Analysis of liquid ensembles for enhancing the performance and accuracy of liquid state machines. *Frontiers in neuroscience*, 13:504, 2019.

[284] Tadashi Yamazaki and Shigeru Tanaka. The cerebellum as a liquid state machine. *Neural Networks*, 20(3):290–297, 2007.

[285] Wolfgang Maass. Liquid state machines: motivation, theory, and applications. *Computability in context: computation and logic in the real world*, pages 275–296, 2011.

[286] Yong Zhang, Peng Li, Yingyezhe Jin, and Yoonsuck Choe. A digital liquid state machine with biologically inspired learning and its application to speech recognition. *IEEE transactions on neural networks and learning systems*, 26(11):2635–2649, 2015.

[287] Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

[288] Robert G Gallager. *Information theory and reliable communication*, volume 588. Springer, 1968.

[289] Robert B Ash. *Information theory*. Courier Corporation, 2012.

[290] Meheroo Jussawalla and Yale M Braunstein. Information theory and telecommunications: A review. *Between Communication and Information*, pages 137–169, 2017.

[291] Anupama Mishra, Neena Gupta, and BB Gupta. Defense mechanisms against ddos attack based on entropy in sdn-cloud using pox controller. *Telecommunication systems*, 77:47–62, 2021.

[292] John C Baez and Blake S Pollard. Relative entropy in biological systems. *Entropy*, 18(2):46, 2016.

[293] Chang Francis Hsu, Sung-Yang Wei, Han-Ping Huang, Long Hsu, Sien Chi, and Chung-Kang Peng. Entropy of entropy: Measurement of dynamical complexity for biological systems. *Entropy*, 19(10):550, 2017.

[294] Soheil Keshmiri. Entropy and the brain: An overview. *Entropy*, 22(9):917, 2020.

[295] Christopher W Lynn, Eli J Cornblath, Lia Papadopoulos, Maxwell A Bertolero, and Danielle S Bassett. Broken detailed balance and entropy production in the human brain. *Proceedings of the National Academy of Sciences*, 118(47):e2109889118, 2021.

[296] Adam C Hull and J Bruce Morton. Activity-state entropy: A novel brain entropy measure based on spatial patterns of activity. *Journal of Neuroscience Methods*, 393:109868, 2023.

[297] Fali Li, Lin Jiang, Yuanyuan Liao, Yajing Si, Chanli Yi, Yangsong Zhang, Xianjun Zhu, Zhenglin Yang, Dezhong Yao, Zehong Cao, et al. Brain variability in dynamic resting-state networks identified by fuzzy entropy: a scalp eeg study. *Journal of neural engineering*, 18(4):046097, 2021.

[298] Viktor Jirsa and Hiba Sheheitli. Entropy, free energy, symmetry and dynamics in the brain. *Journal of Physics: Complexity*, 3(1):015007, 2022.

[299] Robin L Carhart-Harris. The entropic brain-revisited. *Neuropharmacology*, 142:167–178, 2018.

[300] Robin Lester Carhart-Harris, Robert Leech, Peter John Hellyer, Murray Shanahan, Amanda Feilding, Enzo Tagliazucchi, Dante R Chialvo, and David Nutt. The entropic brain: a theory of conscious states informed by neuroimaging research with psychedelic drugs. *Frontiers in human neuroscience*, page 20, 2014.

[301] Sidath Rankaduwa and Adrian M Owen. Psychedelics, entropic brain theory, and the taxonomy of conscious states: a summary of debates and perspectives. *Neuroscience of consciousness*, 2023(1):niad001, 2023.

[302] Joshua S Richman, Douglas E Lake, and J Randall Moorman. Sample entropy. In *Methods in enzymology*, volume 384, pages 172–184. Elsevier, 2004.

[303] Yang Dan and Mu-ming Poo. Spike timing-dependent plasticity of neural circuits. *Neuron*, 44(1):23–30, 2004.

[304] Natalia Caporale and Yang Dan. Spike timing–dependent plasticity: a hebbian learning rule. *Annu. Rev. Neurosci.*, 31:25–46, 2008.

[305] Henry Markram, Wulfram Gerstner, and Per Jesper Sjöström. A history of spike-timing-dependent plasticity. *Frontiers in synaptic neuroscience*, 3:4, 2011.

[306] Sabrina Tazerart, Diana E Mitchell, Soledad Miranda-Rottmann, and Roberto Araya. A spike-timing-dependent plasticity rule for dendritic spines. *Nature communications*, 11(1):4276, 2020.

[307] Dominique Debanne and Yanis Inglebert. Spike timing-dependent plasticity and memory. *Current Opinion in Neurobiology*, 80:102707, 2023.

[308] Sander M Bohte, Han La Poutré, and Joost N Kok. Unsupervised clustering with spiking neurons by sparse temporal coding and multilayer rbf networks. *IEEE Transactions on neural networks*, 13(2):426–435, 2002.

[309] Jason K Eshraghian, Max Ward, Emre O Neftci, Xinxin Wang, Gregor Lenz, Girish Dwivedi, Mohammed Bennamoun, Doo Seok Jeong, and Wei D Lu. Training spiking neural networks using lessons from deep learning. *Proceedings of the IEEE*, 2023.

[310] Abigail Morrison, Markus Diesmann, and Wulfram Gerstner. Phenomenological models of synaptic plasticity based on spike timing. *Biological cybernetics*, 98:459–478, 2008.

[311] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004.

[312] Gang Niu, Bo Dai, Makoto Yamada, and Masashi Sugiyama. Information-theoretic semi-supervised metric learning via entropy regularization. *Neural computation*, 26(8):1717–1762, 2014.

[313] Terry Bossomaier, Lionel Barnett, Michael Harré, Joseph T Lizier, Terry Bossomaier, Lionel Barnett, Michael Harré, and Joseph T Lizier. *Transfer entropy*. Springer, 2016.

[314] Dennis Joe Harmah, Cunbo Li, Fali Li, Yuanyuan Liao, Jiuju Wang, Walid MA Ayedh, Joyce Chelangat Bore, Dezhong Yao, Wentian Dong, and Peng Xu. Measuring the non-linear directed information flow in schizophrenia by multivariate transfer entropy. *Frontiers in computational neuroscience*, 13:85, 2020.

[315] Paolo Victor Redondo, Raphaël Huser, and Hernando Ombao. Measuring information transfer between nodes in a brain network through spectral transfer entropy. *arXiv preprint arXiv:2303.06384*, 2023.

[316] Ning Yin, Haili Wang, Zhaoya Wang, Keke Feng, Guizhi Xu, and Shaoya Yin. A study of brain networks associated with freezing of gait in parkinson's disease using transfer entropy analysis. *Brain Research*, 1821:148610, 2023.

[317] Ernest Montbrió, Diego Pazó, and Alex Roxin. Macroscopic description for networks of spiking neurons. *Physical Review X*, 5(2):021028, 2015.

[318] Liang Chen and Sue Ann Campbell. Exact mean-field models for spiking neural networks with adaptation. *Journal of Computational Neuroscience*, 50(4):445–469, 2022.

[319] Benoit Duchet, Christian Bick, and Áine Byrne. Mean-field approximations with adaptive coupling for networks with spike-timing-dependent plasticity. *Neural computation*, 35(9):1481–1528, 2023.

[320] Stephen Coombes. Next generation neural population models. *Frontiers in Applied Mathematics and Statistics*, 9:1128224, 2023.

[321] Eugene M Izhikevich. Which model to use for cortical spiking neurons? *IEEE transactions on neural networks*, 15(5):1063–1070, 2004.

[322] Fanjun Li, Xiaohong Wang, and Ying Li. Effects of singular value spectrum on the performance of echo state network. *Neurocomputing*, 358:414–423, 2019.